STRAIGHT ROADS EXTRACTION BY FUSION OF SPOKE WHEEL AND REGION GROWING

Xiangguo Lin [a,b*], Jixian Zhang [a], Zhengjun Liu[a], Jing Shen[b]

[a] Key Laboratory of Mapping from Space of State Bureau of Surveying and Mapping, Chinese Academy of Surveying and Mapping, No.16, Beitaiping Road, Haidian District, Beijing 100039, China - (linxiangguo@gmail.com, zhangjx@casm.cn, zjliu@casm.cn)
[b] School of Resources and Environment Science, Wuhan University, 129, Luoyu Road, Wuchang District, Wuhan 430079, China - shenjing00@163.com

KEY WORDS: Semi-automatic; road extraction; spoke wheel; region growing

ABSTRACT:

Semi-automatic methods are regarded as more practical than automatic methods to obtain and/or update the vector road data. But road tracking methods lose their efficiencies when they are applied to track a long straight roads step by step compared to manual digitalizing. So an efficient semi-automatic method is proposed in this paper. Particularly, spoke wheel algorithm and region growing algorithm are combined together to extract a rectangle corresponding to a short road segment. That above process is performed at each pixel along a line segment between two seed points, which derives a number of rectangles. Then statistical analysis of histograms of directions and widths of the obtained rectangles is made to determine an optimal rectangle. At last, a precise road model is determined by the above optimal rectangle. Extensive experiments show that our method is capable of efficiently extracting a straight long road with a high spatial accuracy, even various types of noises existing.

1. INTRODUCTION

With the increasing availability of commercial high-resolution imaging sensors, automatic/semi-automatic interpretation tools are needed to efficiently extract road networks with reliable spatial accuracy. Actually, automatic/semi-automatic extraction of road networks from remotely sensed imagery is not only meaningful for cartography and topography, but also significant for various applications of geodata such as automatically aligning two spatial datasets or automated vehicle navigation (Lin, 2009). Despite the fact that much work on automatic approaches for road extraction has taken place, the desired high level of automation could not be achieved yet (Baumgartner et al., 2002). The main problem of a fully automatic approach is that it needs some strict hypothesis of road characteristics, but road properties vary considerably with ground sampling distances (GSD), road types, and densities of surrounding objects, light conditions etc (Zhou et al., 2006). Therefore, the quality of automatic extraction is usually insufficient for practical applications. On the other hand, semi-automatic methodologies are considered to be a good compromise between the fast computing speed of a computer and the efficient interpretation skills of a human operator (Zhou et al., 2006), and quite a number of promising approaches for semi-automatic road extraction have been proposed so far (Mena, 2003). For example, optimal search methods, which are often realized by dynamic programming (Gruen and Li, 1995) or snakes(Gruen and Li, 1995; Niu, 2006), are frequently applied to find or determine an optimal trajectory between manually selected seed points. In these models, geometric and radiometric characteristics of roads are integrated by a cost function or an "energy" function. Then the road extraction is equivalent to seeking the global energy minimum. However, it is hard to define the reasonable "energy" function for each image. Promisingly, Hu et al. (2007) combined a spoke wheel operator, used to detect road surfaces, and a toe-finding algorithm, utilized to determine the road direction, to trace roads. But the extracted results are the inscribed lines rather than the centerlines needed for cartography, and that algorithm is not very robust to noises. Another more practical methodology is road tracking by template matching. Particularly, McKeown and Denlinger (1988) presented a road tracking method based on the intensity profile correlation of road cross sections to follow the direction of a road. Vosselman and Knecht (1995), Baumgartner et al. (2002) imposed the profile matching by using least squares template matching and Kalman filter. Zhou et al. (2006) used two profiles, one perpendicular to the road direction and the other parallel to the road direction, to enhance the robustness of the tracker and applied extended Kalman filter and particle filter to solve profile matching issues for road tracking; this method also integrates the online learning with novelty detection to adapt to the road features change (Zhou et al., 2007). Kim et al. (2004) employed a rectangular template to track roads by least squares template matching, and the road path is modeled as similarity transform. Zhao et al. (2002) used rectangular template matching on the basis of a classified imagery. Hu et al. (2004) presented a road tracker using a piecewise parabolic model and least squares template matching. Lin et al. (2010) utilized the distance transformation to erode the image noises in order to enhance the reliability of template matching. Despite the above road tracking methods can extract the road width, the road direction besides a road centerline, the experiments demonstrate that existing step-by-step tracking method is more preponderant for smooth and curving long ribbon roads than for straight roads (Lin et al., 2009), because it is inefficient to track the straight roads compared to the curving roads. However, high efficiency is an important criterion in semi-automatic road tracker performance evaluation (Hu et al., 2004; Zhou et al., 206), meanwhile the straight roads account for most of road networks, especially in the urban areas (Price, 2000; Haverkamp, 2002). In this sense,

* Corresponding author. linxiangguo@gmail.com

the efficiency of road network extraction is determined by the efficiency of straight roads extraction.

Considering the importance of extracting the straight roads, many methods had been proposed. It is well known that a short road segment on very high resolution (VHR) remotely sensed imagery has a very large rectangularity (Hu et al., 2007). That character has been utilized to detect the whole road networks by many researchers such as Nagao and Matsuyama (1980) Song and Civco (2004), Shi and Zhu (2002), and Hu et al. (2007) etc. Meanwhile, edge detection algorithms are also utilized to extract straight roads. For instance, radon transform was employed to locate the roadsides and to measure the width of a road (Zhang and Couloigner, 2007). However, the above automatic methods are sensitive to various types of noises such as occlusion of vehicle and shadow of trees, and they can not detect precise road width and road direction which are essential for depicting a straight road in practice. In this paper, a semi-automatic method is proposed to extract a straight road.

As mentioned above, a short road segment is equivalent to a rectangle (Hu et al., 2007). If a road is not seriously affected by noises, spoke wheel algorithm (Hu et al., 2007) is capable of reliably extracting a rectangle at a road point. That rectangle can supply enough information such as direction and width to depict a whole straight road. That means the spoke wheel algorithm is helpful to extract a whole road. However, spoke wheel loses its advantages once encountering the image noises such as traffic jam and shadow of tall buildings, and the VHR image is greatly impacted by various types of noises. Fortunately, fusion of the spoke wheel algorithm and region growing method can overcome the above noises in this paper. Once a starting point and an end point are selected by a human operator, a number of rectangles will be obtained if the above improved spoke wheel algorithm is run at each pixel along the line segment between the starting point and the end point. Then analysis of histogram of rectangles is helpful to determine the optimal rectangle responding to a straight road. Once the optimal rectangle is given, it means width and direction of the road are determined, and the starting point and the end point can be adjusted based on the precise road direction. At last, a straight road is accurately depicted by the starting point, the end point and the width. Overall, the spoke wheel algorithm, the region growing algorithm are combined to extract a precise rectangle corresponding to a short road segment, and maximum likelihood estimation is employed to find the optimal rectangle corresponding to a straight road.

The organization of this paper are: the methods are described in Section II, experiments and performance evaluation are presented in Section III, and results are summarized in Section IV.

2. METHOD

2.1 Spoke wheel algorithm

Theoretically, each road pixel has a local homogeneous region, which is anisotropic and directionally rectangular. That is, along some directions, the branches of the local homogeneous region are approximately rectangular. Hu et al. (2007) proposed the spoke wheel operator to delineate the boundary of the homogeneous region. Hu's spoke and the spoke wheel SW are used for detecting a footprint of a pixel p, as shown in Fig. 1a. A spoke is a line segment with a length of m (a preset value) pixels, and m = 90 pixels in Fig. 1a. A spoke wheel is a sequence of spokes $S(\varphi_i, m)$ (i =0,…, 4 n −1) with common initial point p and evenly spaced angles $\varphi_i = \pi i / 2n$, and n =18 in Fig. 1a. The set of pixels in a spoke wheel SW centered at the pixel p with 4 n spokes is denoted by $SW(p, n, m)$. And the intersection between a spoke and a road edge provides useful information to determine the local homogeneous region around a pixel. However, we do not know the exact location of a road edge in advance. To search for the intersection of a spoke and the edge of a road, we start from p, move in the direction of the spoke, and observe the absolute intensity differences between p and the pixels along the spoke. The differences are small when the pixels are near to p; however, they may become larger when the pixels are far away from p. Let S_i be the spoke at pixel p. The cutting point, denoted by C_i, on S_i is the first pixel such that

$$| I(C_j) - I(p) | \geq \sigma_i \quad (j = 0, \ldots, m, \ 0 \leq i < 4n) \quad (1)$$

where σ_i is the intensity standard deviation on $SW(p, n, m)$. Therefore, the threshold in Equation 1 is adaptive. Subsequently, we connect the cutting points on all spokes around a pixel p in a counterclockwise direction, which results in a closed polygon. This represents the footprint of the pixel p, denoted by $F(p)$, and its corresponding minimal oriented bounding box (Rosin, 1999) is denoted by $MOBB$. We threshold the ratio of its area to that of its $MOBB$. A footprint $F(p)$ is nearly rectangular if

$$\frac{Area(F(p))}{Area(MOBB)} \geq 85\% \quad \&\& \quad \frac{length\ of\ longer\ edge\ of\ MOBB}{length\ of\ shorter\ edge\ of\ MOBB} > 2.0 \quad (2)$$

both hold. The thresholds of 85% and 2.0 were selected by experiments in Hu et al. (2007). If the road surfaces are homogenous, the derived $MOBB$ may best match the road segment, as shown in Fig. 1b. Oppositely, if the road surfaces are not, there may be no $MOBB$ obtained at all, as shown in Fig. 1c.

2.2 Our integration strategy

Region growing (Adams and Bischof, 1994) was employed to decrease the side-effects of image noises on spoke wheel algorithm. Actually, there are three key elements in region growing, namely: a seed point, a similarity threshold or criteria that a region is grown from the seed point to adjacent points depending on, and the connectivity of pixels (4-connected neighborhood or 8-connected neighborhood). In this paper, we take the above pixel p as the seed point, the standard deviation of grey values in $F(p)$ is regarded as similarity threshold value, and 8-connected neighborhood is adopted for our pixels

adjacent relationship. The above process derives a new connected region, denoted as $R(p)$. Let S_i be the spoke at pixel p. The cutting point $C_{i,j}$ on S_i is redefined as the first pixel such that

$$C_{i,j} \cap R(p) \neq \phi \quad (j = m, m\text{-}1, \ldots, 0, 0 \leq i < 4n) \quad (3)$$

Subsequently, we connect the cutting points on all spokes around a pixel p in a counterclockwise direction, which also results in a closed polygon. Replace $F(p)$ with the newly obtained region, and its corresponding minimal oriented bounding box is also denoted as $MOBB$, as shown in Fig. 1e and 1g. As can be seen, the improved spoke wheel is robust to noises, and the newly derived $MOBB$ matches a short road segment.

2.3 The general framework

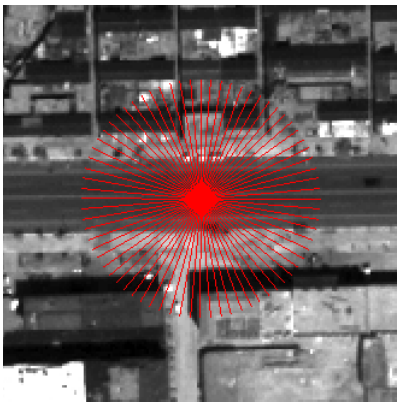
A straight road can be modelled by a starting point, an end point, road width. The basic idea of our approach is that two initial seed points are selected by human operator, precise road width and road direction are measured by the computer, and adjustment of the seed points is made to get precise road model. Thus a framework is designed to extract straight roads as follows:

(1) A human operator detects a straight road and consecutively clicks the mouse two times at the beginning and end of a road respectively, which indicates two seed points, the starting point P_{start} (x_{start} , y_{start}) and the end point P_{end} (x_{end} , y_{end}), where (x_{start} , y_{start}) and (x_{end} , y_{end}) are the coordinates of the two seed points. Moreover, a corresponding line segment is derived by P_{start} and P_{end} , denoted as $S_{segment}$ whose length is denoted as L_{road} , and L_{road} is calculated by:

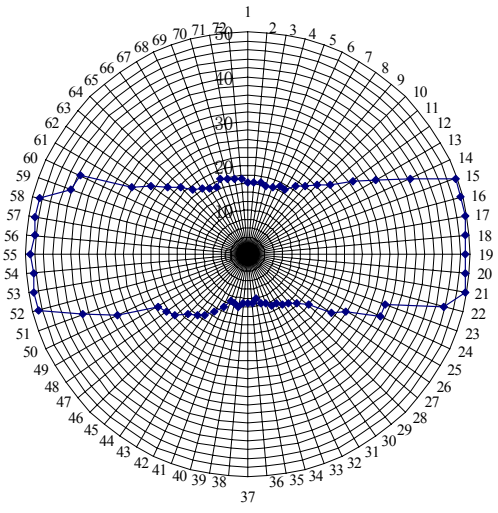
$$L_{road} = \text{Floor}(\sqrt{(x_{start} - x_{\text{end}})^2 + (y_{start} - y_{end})^2}) \quad (4)$$

Note that the direction of $S_{segment}$ is arctangent(P_{start} P_{end}) (0< arctangent(P_{start} P_{end}) $\leq 2\pi$).

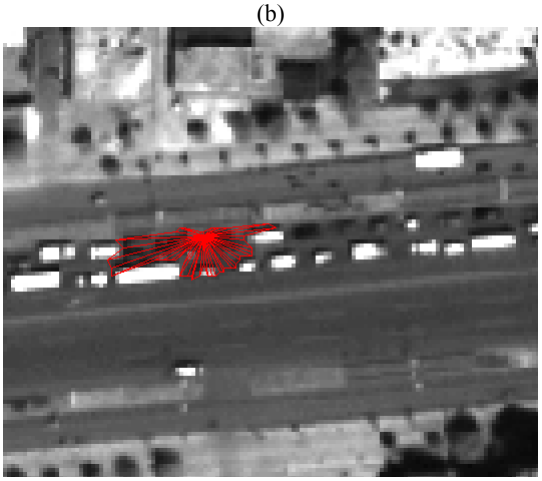
(2) Obtain a seed point at each pixel of $S_{segment}$, denoted as P_k (x_k , y_k)(k =0,…, L_{road}).



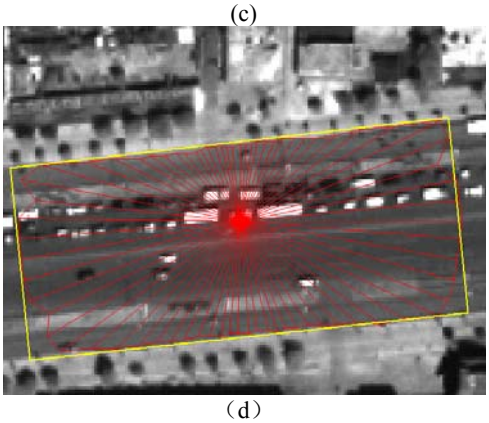
(a)



(b)



(c)



（d）

Fig. 1 (a) A spoke wheel. (b) Result of a spoke wheel algorithm operated on a road image corresponding to (a). (c) Side-effects of occlusion on the spoke wheel algorithm. (d) A subset of image seriously impacted by occlusion of traffic jam and the result of our improved spoke wheel algorithm operated on the image

(3) Take each P_k (x_k , y_k)(k =0,…, L_{road}) as a seed and run the above improved spoke wheel algorithm, if a minimal oriented bounding box is obtained, denoted as $MOBB_k$ whose width is w_k , direction and intercept on the y axis of the line through the two midpoints of shorter sides is θ_k (0< $\theta_k \leq 2\pi$) and b_k respectively. Then add (θ_k , w_k , b_k) into a "vote box". Note that the line through the two midpoints of shorter sides of a rectangle may have two alternatives, but the one with minimal angle with arctangent(P_{start} P_{end}) is selected.

(4) Obtain the total number of minimal oriented bounding box in the "vote box", denoted as T_{MOBB} . In this circumstance:

- If T_{MOBB} =0, then return the control back to the human operator;
- If T_{MOBB} =1, the corresponding information of the minimal oriented bounding box is regard as the road's width w , direction θ and intercept b .then go to step (7);
- If T_{MOBB} >1, then go to step (5).

(5) Obtain the optimal road direction by the following procedure:

① Determine the block size to build the histogram of directions in the above "vote box", denoted as S_{block} , and S_{block} is set to 1° in this paper;

② Find the maximum value and the minimum value of direction values in the "vote box", denoted as $\max(\theta)$ and $\min(\theta)$ respectively, where $\min(\theta) \leq (\theta_k) \leq \max(\theta)$ (k =0,…, T_{MOBB});

③ Determine the total number of blocks, denoted as T_{block} , and T_{block} is calculated by:

$$T_{block} = \text{Ceiling}((\max(\theta) - \min(\theta)) / S_{block}) \quad (5)$$

④ Determine the number of minimal oriented bounding boxes in each block, and find the block who has the largest amount of minimal oriented bounding boxes, denoted as O_θ ;

⑤ Put each minimal oriented bounding box in O_θ into another "vote box", and determine the number of minimal oriented bounding boxes, denoted as T'_{MOBB} . In this circumstance:

- If T'_{MOBB} =1, the corresponding information of the minimal oriented bounding box is regard as the road's width w , direction θ and intercept b , then go to step (7);
- If T'_{MOBB} >1, then go to step (6).

(6) Obtain the optimal road width by the following procedure:

① Determine the block size to build the histogram of widths in the above newly obtained "vote box", denoted as S'_{block} , and S'_{block} =1 in this paper;

② Find the maximum value and the minimum value of width values in the "vote box", denoted as $\max(w)$ and $\min(w)$ respectively, where $\min(w) \leq w_k \leq \max(w)$ (k =0,…, T'_{MOBB});

③ Determine the total number of blocks, denoted as T'_{block} , and T'_{block} is calculated by:

$$T'_{block} = \text{Ceiling}((\max(w) - \min(w)) / S'_{block}) \quad (6)$$

④ Determine the number of minimal oriented bounding boxes in each block, and find the block who has the largest amount of minimal oriented bounding boxes, denoted as O_w ;

⑤ Find the optimal minimal oriented bounding box whose width has a minimal difference with the median of O_w , and the corresponding information of the optimal minimal oriented bounding box is regard as the road's width w , direction θ and intercept b .

(7) Update the road model by (w , θ , b) as follows:

Suppose a line pass through P_{start} (x_{start} , y_{start}) and it is perpendicular to the line $y = \tan(\theta) + b$, the foot of the perpendicular is regarded as the true starting point, denoted as P'_{start} (x'_{start} , y'_{start}). Similarly, suppose a line pass through P_{end} (x_{end} , y_{end}) and it is perpendicular to the line $y = \tan(\theta) + b$, the foot of the perpendicular is regarded as the true end point, denoted as P'_{end} (x'_{end} , y'_{end}). Then the true road model is determined by the starting point P'_{start} (x'_{start} , y'_{start}), the end point P'_{end} (x'_{end} , y'_{end}), plus the road width w .

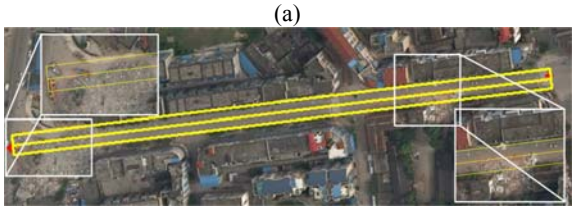
3. EXPERIMENTS AND PERFORMANCE EVALUATION

We developed a testing platform based on the above method to extract straight roads from VHR images, and attempted to evaluate how fast (efficiency) and how accurate (spatial accuracy) our method is compared to a fully manual digitizing. Particularly, the efficiency is evaluated by comparing the total time consumed in digitizing the same road (Hu et al., 2004), and spatial accuracy is evaluated by comparing length, width and direction between the results of the two methods. In the digitizing process of our method, two seed points are input through the interface by clicking the mouse at a reasonable scale. But for the manual annotation, three seed points are input as follows: a human operator enters a road segment with two seed points on A' and B' whose axis joining the points defining one road sideline A'B', which indicates road direction arctangent(A'B') and road length l . l is the equal to the distance between point A' and point B'. Then the following

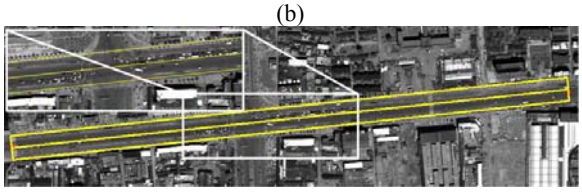
third seed on C, on the other roadside, defines the road width w. w is equal to the distance between the point C and the line A'B'. As a result, the above three points can also derive a rectangle A'B'B"A" with width w and length l, as shown in Fig. 2.

A great number of aerial and satellite images of different GSD have been tested, and four images are displayed in Fig. 3. In fact, the four images are impacted by various types of noises. Particularly, Fig. 3a shows an arterial road on an aerial photograph covered in an urban area, and the road is impacted by occlusion of vehicles and shadow of trees. Fig. 3b shows a residential road on an aerial image covered in a suburban area, and the road is seriously polluted by surrounding collapsed buildings. Fig. 3c shows an arterial road on a QuickBird panchromatic image covered in an urban area, and the road is seriously impacted by occlusion of traffic jam. Fig. 3d shows an arterial road on a QuickBird panchromatic image covered in an urban area, and the road is seriously occluded by colonnades.

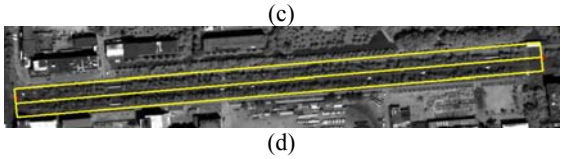
(a)



(b)



(c)



(d)

Fig. 3 Semi-automatic straight road extraction from VHR images. (a) Extraction of an arterial road from an aerial photograph, ground sampling distance = 0.2 m/pixel, image size = 1585×2120 pixels. (b) Extraction of a residential road from an aerial photograph, ground sampling distance = 0.2 m/pixel, image size = 1346×450 pixels. (c) Extraction of an arterial road from a QuickBird panchromatic image, image size = 1844×521 pixels. (d) Extraction of an arterial road from a QuickBird panchromatic image, image size = 395×185 pixels.

In the experiments, the parameter m, length of a spoke refers to Table 1, and Fig. 4 shows the corresponding histogram of widths. The corresponding extracted results of our method are superimposed on the original images, as Fig. 3 shown. Visual inspection tells that the extracted rectangles are reasonable and accurate even the many noises exiting. Table 1 shows the statistical result on testing the efficiency and spatial accuracy of our semi-automatic road extraction method. In general, comparing to the fully manual annotation, our method has a higher efficiency and a comparable spatial accuracy.

Image No.	Method	Length of a spoke (pixels)	Time cost (second)	Length (pixels)	Width (pixels)	Direction difference (°)
(1)	Manual	-	7	2261	77	-
	Semi-automatic	100	6	2256	77	4
(2)	Manual	-	5	871	46	-
	Semi-automatic	60	4	879	44	3
(3)	Manual	-	6	761	93	-
	Semi-automatic	120	5	764	95	2
(4)	Manual	-	5	1067	49	-
	Semi-automatic	70	3	1062	51	6

Table 1. Comparison of semi-automatic and manual methods

Our method's better performance comes from its several merits. First, it utilizes the fast computing speed of a computer. On state-of-the-art computers, the volume of computation of our method is not a key constraint. Automatic measurement may be completed as soon as the mouse click is operated for most of the straight roads. Second, our method can be operated at a coarsely scale. A human operator can input the two necessary seed points at a low level of image pyramid for our method. But, for fully manual annotation, a human operator has to input the seed points at a finer scale so that the spatial accuracy is guaranteed, and additional operations such as zoom and pan may be needed for a long road to get a precise result on a VHR image. Comparably, our method is not sensitive to the raw spatial resolution as manual digitalizing does, which means a time saving in initialization compared to manual digitalizing. Third, our method utilizes only one parameter, the length m of a spoke, needed human involvement before extracting, but the extracted result is not very sensitive to value of m. For the above four tests, m is approximately equal to the true road width multiplying by 1.25. In fact, the road with is relevant to road types in practice. In this sense, m can be set to a constant for each type of roads. For example, m is 110 pixels for arterial roads and 70 pixels for streets on aerial images. Last but not least, our method utilizes the maximum likelihood estimation. In fact, only a precise rectangle is enough to depict a whole

road for our semi-automatic method. Even a straight road is seriously impacted by noises, but not all parts of a straight are polluted. That means that our fusion method is capable of extracting the rectangle that can derive inform to depict a whole road if some a part of the road is satisfied. Certainly, more rectangles are helpful to get a better result. Then histogram analysis is employed to get the most precise result. Those advantages of our method guarantee the high efficiency and precise spatial accuracy of the results.

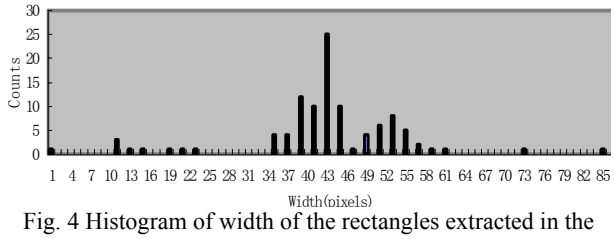


Fig. 4 Histogram of width of the rectangles extracted in the second test image

4. CONCLUSION

This paper presents a semi-automatic method for extracting straight roads from VHR images. On the basis that a short road segment is equivalent to a rectangle, the fusion of spoke wheel algorithm and region growing is employed to extract the rectangle corresponding to a short road segment even varying image noises existing. Meanwhile, histogram analysis is employed to obtain the most precise rectangle from the extracted results along the line segment of two input seed points. Various types of aerial and satellite images are used to test our method. The results demonstrate that our method is capable of efficiently extracting the straight roads with precise results. Future work will focus on improvement of our method to make it feasible to curve roads.

ACKNOWLEDGEMENTS

This research is funded by the National Key Basic Research and Develop Program under Grant No. 2006CB701303 and Project for Young Scientist Fund sponsored by the National Natural Science Foundations of China under Grant No. 40401037.

REFERENCES

Adams, R., Bischof, L., 1994. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 641-647.

Baumgartner, A., Hinz, S., Wiedemann, C., 2002. Efficient methods and interfaces for road tracking. *International Archives of Photogrammetry and Remote Sensing*, vol. 34, Part 3B, pp. 309-312.

Kim, T., Park, S.R., Kim, M.G., Jeong, S., Kim, K.O., 2004. Tracking road centerlines from high resolution remote sensing images by least squares correlation matching. *Photogrammetric Engineering and Remote Sensing*, vol. 70, no. 12, pp. 1417-1422.

Lin, X.G., Liu, Z.J., Zhang, J.X, Shen, J., 2009. Combing multiple algorithms for road network tracking from multi-source remotely sensed imagery: a practical system and performance evaluation. *Sensors*, vol. 9, no. 2, pp.1237-1258.

Lin, X.G., Liu, Z.J., Zhang, J.X, Shen, J., 2010. Semi-automatic road tracking by interlaced template matching. *International Journal of Remote Sensing*, (in press)

Gruen, A., Li, H., 1995. Road extraction from aerial and satellite images by dynamic programming. *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 50, no. 4, pp. 11-20.

Gruen, A., Li, H., 1997. Semi-automatic linear feature extraction by dynamic programming and LSB-Snakes. *Photogrammetic Engineering and Remote Sensing*, vol. 63, no. 8, pp. 985-995.

Haverkamp, D., 2002. Extracting straight road structure in urban environments using IKONOS satellite imagery. *Optical Engineering*, vol. 41, no. 9, pp. 2107-2110.

Hu, X., Zhang, Z., Tao, C., 2004. A robust method for semi-automatic extraction of road centerlines using a piecewise parabolic model and least squares template matching. *Photogrammetric Engineering and Remote Sensing*, vol. 70, no. 12, pp. 1393-1398.

Hu, J., Razdan, A., Femiani, J.C. et al., 2007. Road network extraction and intersection detection from aerial images by tracking road footprints. *IEEE Transaction on Geoscience and Remote Sensing*, vol. 45, no. 12, pp. 4144-4157.

Mckeown, D.M., Denlinger, J.L., 1988. Cooperative methods for road tracing in aerial imagery. In *Proceedings of the IEEE Conference in Computer Vision and Pattern Recognition*, 1988, pp. 662–672.

Mena, J.B., 2003. State of the art on automatic road extraction for GIS update: a novel classification. *Pattern Recognition Letters*, vol. 24, no. 16, pp. 3037- 3058.

Nagao, M., Matsuyama, T., 1980. A Structural Analysis of Complex Aerial Photographs. In: Nadler, M. (Ed.), Advanced Application in Pattern Recognition, Plenum Press, New York, Vol. 1, 1980, pp. 1-199.

Niu, X., 2006. A semi-automatic framework for highway extraction and vehicle detection based on a geometric deformable model. *ISPRS Journal of Photogramemetry and Remote Sensing*, vol. 61, pp. 170-186.

Shi, W., Zhu, C., 2002. The line segment match method for extracting road network from high-resolution satellite images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 2, pp. 511-514.

Song, M., Civco, D., 2004. Road extraction using SVM and image segmentation. *Photogrammertric Engineering and Remote Sensing*, vol. 70, no. 12, pp. 1365-1371.

Rosin, P.L., 1999. Measuring rectangularity. *Machine Vision Application*, vol. 11, no. 4, pp. 191-196.

Price, K., 2000. Urban Street Grid Description and Verification. In *5th IEEE Workshop on Applications of Computer Vision*, 2000, pp. 148–154.