International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol. 10, No. 7 (2017), pp. 61-72
http//dx.doi.org/10.14257/ijsip.2017.10.7.06

A Fast Target Locating Method for Remote Sensing Images Based on Line Features

Xinjian Wang*, Guangchun Luo, Ke Qin, Aiguo Chen and Ling Tian

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
wangxinjian_lw@163.com

Abstract

Target location is an effective approach to mass information filtering for remote sensing images. In this paper, a simple and fast target location method is proposed based on line features for pre-registrated remote sensing images. Firstly, a line model of the specific target to be located is generated by manually labeling line features in the historical images containing the target. Secondly, straight lines are extracted from the remote sensing images to be searched. Thirdly, the target is located by matching line features in the line model and extracted lines in the remote sensing image. The optimal matching position is obtained by offset-accumulations similar with a voting mechanism. Numerous experimental results reveal that our proposal is effective and efficient for target locating with pixel-level accuracy in remote sensing images. Our proposal is also robust for partially occlusion in remote sensing images with small registration error.

Keywords*: Fast target locating , Line matching , Offset accumulator , partial occlusion*

1. Introduction

With the fast development of sensors and satellite technology, soaring amounts of remote sensing data are obtained with high temporal, spatial and spectral resolution. However, users are only interested in the target which occupies a small region in the remote sensing image. The task of target location technique is to locate a specific target in the image based on a target model. Therefore, the region of interest is located and extracted, which releases the burden on storage capacity and computational resource and thus improves the utilization rate of remote sensing data.

The target location technique is similar to the image registration technique in finding the relative spatial transform parameters between the target image patch and the remote sensing image. The state-of-the-art target location techniques are generally divided into two categories [1-4]. The first category calculates the similarity of the target model and the searching area in the remote sensing image based on grayscale. Techniques of this category have the following disadvantages: 1) it is unlikely to get an accurate result when the image texture is not rich or the signal noise ratio (SNR) is low; 2) target templates are required to build the target model. The second category is based on features. Techniques of this category match features of the target with features extracted from the image. Generally, points [5-6], lines [7], contours [8-10] and other features [11] are used to describe the target. Compared with the first category which adopts target template, techniques in this category require smaller storage capacity. Furthermore, features describe the target more effectively and efficiently as a high-level representation of the target, which meets the requirements of high computational efficiency in real time applications.

SIFT is one of the most famous point features widely used in image auto-registration. However, in weakly textured regions, such as desert, farmland and ponds, it is difficult to extract enough feature points. To address this problem, in this paper, we adopt line

ISSN: 2005-4254 IJSIP

features to describe the target. In remote sensing images, most ground targets of interest are man-made facilities which usually contain a large number of line features, such as runways in airports, berths in ports and roads in cities [12]. Compared with point features, line features are much more stable an robust to image conditions. Besides, only two endpoints are needed to describe a line feature, which is simpler than SIFT with a 128-dimensional feature vector [13].

After extracting line features from the target image patch and the remote sensing image to be searched, the issue of target location is transformed into a matching problem between line sets. Early works on line matching exist but suffer from the following two issues: 1) Due to the influence of image noise and cloud cover, it is hard to extract intact lines from the images, which results in the decrease of matching precision or even failure. 2) These methods consider the similarities of local properties of features and thus a global topological optimal cannot be achieved [14]. Some other methods adopt the concept of global optimal algorithm but are computationally intensive.

This paper aims at a fast locating technique for a specific ground target in a remote sensing image. Given that geometric rectification and registration is imposed on the remote sensing images in advance, only the offset and small rotation is to be determined in this location technique. We give prior consideration to robustness and computational efficiency when designing this location technique.

The remaining part of this paper is organized as follows. The line model of the target to be located is described in Section 2. The details of this target location technique are described in Section 3. Experimental results are shown in Section 4, which gives detailed analysis on precision, computation time and robustness of this technique. Conclusions of this paper are exposed in Section 5.

2. Line Model

The line model is an accurate description of the target in images. In this target location technique, we obtain accurate line features by manual annotation instead of some automatic line extraction methods. Compared with automatic line extraction methods, manual annotation is able to guarantee the precision of line features in the following two aspects: 1) Automatic line extraction methods ae sensitive to image noise. 2) Manual annotation only focuses on the line features on the target while automatic line extraction methods extract lines of the whole image.3) Manual annotation is able to determine the length of line features.

Latitude and longitude coordinates of two endpoints of a line feature are labeled manually and saved in the line model with no special requirements for their order. The reason why latitude and longitude coordinates are used instead of the image coordinates will be explained later in this paper. Several points should be considered when manually annotating a target in a remote sensing image: 1) Line features should be annotated as obvious line edges on the target. 2) Line features should be evenly distributed in various parts of the target mainly to ensue enough line features are still visible under partial occlusion. A line model of Tianjin Binhai International Airport in China is shown as in Figure 1 (see the red line part).

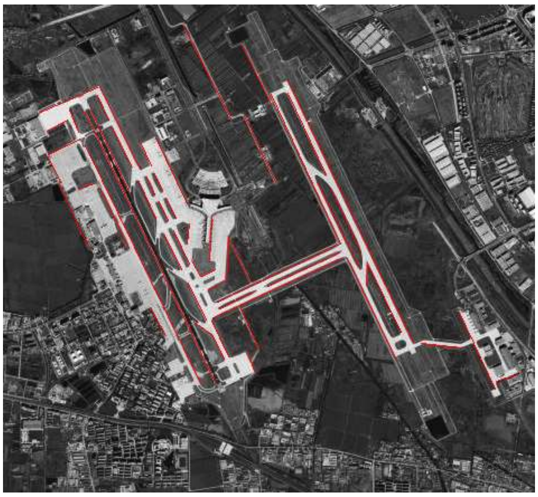


Figure 1. Line Model of an Airport

3. Target Locating

Given that geometric rectification and registration have been done on the input remote sensing image and the transformation between latitude and longitude coordinates and image coordinates is obtained in advance, the task of target location can be described as: given the line model of a specific target and a remote sensing image, judge whether the target exists in the image and locate it if it exists. We judge whether a target exists by comparing the maximum matching rate with an empirically given threshold. The flow chart of our target location technique is as follows. Firstly, a line model is obtained by manual annotation and line features are extracted from the remote sensing image to be searched. Secondly, offset accumulation between the line model and extracted lines are calculated. Lastly, the maximal matching point is found, verified and outputted as the final result.

3.1. Representation of the Line Model

Each line feature in the line model is represented by the latitude and longitude coordinates of two endpoints while the extracted lines from the remote sensing image are represented by image coordinates. To locate the target, the latitude and longitude coordinates in the line model are transformed into image coordinates. Given M line features in the line model $L=\{l_1,l_2,\cdots,l_M\}$, each line feature is represented by a feature vector:

$$l_i=(x_i^s,y_i^s,x_i^e,y_i^e,\theta_i,l_i,l_i^p)\ (i=1,2,\cdots,M)\,.$$

In the feature vector, x_i^s and y_i^s are the image coordinates of the starting point of the feature line while x_i^e and y_i^e are the image coordinates of the ending point. θ_i ($\theta_i\in[0^\circ,180^\circ)$), l_i and l_i^p are the angle, length and pixel length respectively. The difference between l_i and l_i^p is that the former is the straight-line distance of two endpoints and the latter is the pixel summation of the straight line between two endpoints which is usually larger than the former.

3.2. Line Feature Extraction

Existing line detection methods can be roughly divided into Hough transform methods, phase-grouping methods and Kalman filtering methods. In this paper, we adopt a robust line extraction method [15] with the following advantages: (1) Kalman filter is used to track edge points, which can make up the missing details in edge extraction and refinement; (2) high precision and strong anti-noise ability can be achieved; (3) calculation is simple and fast.

We use thisabove mentioned line detection method to extract line features on remote sensing images. Assume N lines (including endpoint coordinates, angles and lengths) are extracted on a $W \times H$ sized remote sensing image, the extracted lines arerepresented as $S = \{s_1, s_2, \cdots, s_N\}$, where $s_j = (x_j^s, y_j^s, x_j^e, y_j^e, \theta_j, l_j, l_j^p)$ $(j = 1, 2, \cdots, N)$ and meanings of parameters are similar to the line mode mentioned earlier.

3.3. Offset Accumulator

As mentioned earlier, the issue of target location is transformed into a matching problem between two line sets. In this subsection, we adopt a voting mechanism to determine the offset between the line set in the line model and the line set extracted from the remote sensing image. Different from the pixel-by-pixel search mechanism, our voting mechanism performs line-to-line matching between the line set in the line model and the line set extracted from the remote sensing image. Given M lines in the line model and N lines extracted from the image, we should conduct matching on $M \times N$ line pairs. It is obvious that most of the line pairs are not parallel. Therefore, to reduce the computation burden, we conduct matching only on quasi-parallel line pairs, which means the difference between the angles of the line pair is within a given threshold.

Here is our idea. We generate the line model by manual annotation from a target image patch. The target image patch is shift to the upper-left corner of the remote sensing image as shown in Figure 2. Similar to Hough voting, we construct an offset accumulator of size $(W_M + W) \times (H_M + H)$. Each point (m,n) in the accumulator is calculated by shifting the line set in the target image patch by (m,n) and summing the matching score of all the quasi-parallel line pairs.

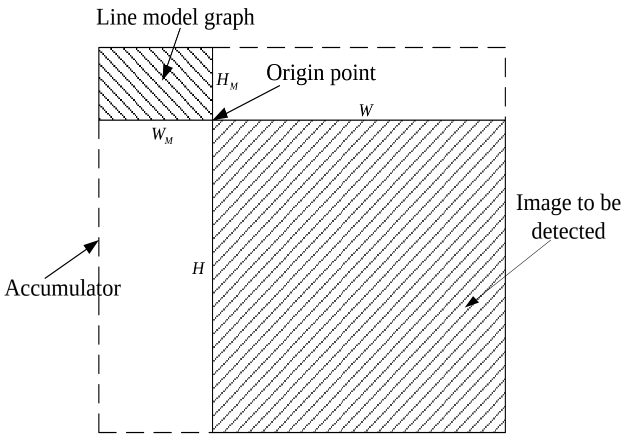


Figure 2. The Combination of Line Model Graph and the Image to be Detected

We select a line feature l_i from the line model Land s_j from the extracted lines S respectively. If the difference between their angles is bigger than a give threshold, these two lines are considered to be parallel, otherwise skip processing. In the next step, we exchange the starting point and end point of s_j to make sure that the directions of l_i and s_j are the same. We first translate l_i from the start point of l_i to the start point of s_j. Therefore, part of l_i and s_j overlap with each other. Then, we shift l_i in the parallel direction pixel-by-pixel and count the overlap pixels of l_i and s_j as shown in Figure 3. The values in the accumulator along the parallel direction are as shown in Figure 4. The length of ab is the difference between the length of l_i and s_j.

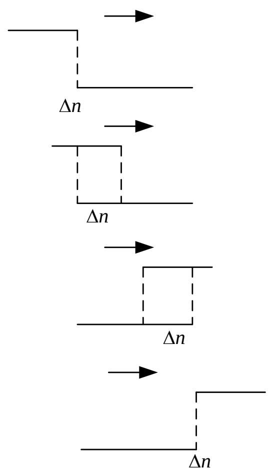


Figure 3. Diagram of Line Overlap Statistics

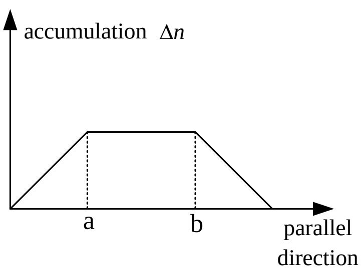


Figure 4. Distribution of the Accumulation of Each Line Pair Overlapping

For multiple line pairs of different directions, multiple corresponding accumulative lines intersect intersects in the accumulator. Theoretically, the intersection point $(x_{\max}, y_{\max})$ is the maximum position in the accumulator. Therefore, the target location can be calculated as $(x_{\max} - W_M, y_{\max} - H_M)$. In fact, due to image noise and the error introduced from extracting lines, the target location often lies within a small neighborhood of the maximum position in the accumulator. The radius of the neighborhood determines the precision of our target locating technique. If the maximum value in the accumulator is smaller than a given threshold, we can conclude that the target doesn't exist in the remote sensing image, *i.e.* location fails. Otherwise, the location $(x_{\max} - W_M, y_{\max} - H_M)$ will be the location result.

3.4. Exception Handling

In practical use, cloud covering and geometric registration error is two challenging factors to our target locating technique.

(1) Cloud covering often happens in the remote sensing images, resulting in occlusion on the concerned target. Fortunately our target locating technique is able to deal with this situation. When the target is occluded, some of the line features in the remote sensing image are not detected and thus their offsets to the model lines will not be accumulated. Therefore, we can still get a maximum accumulative value although it is smaller than the theoretical value. If this maximum accumulative value is greater than a given threshold, the target can still be located correctly. On the other hand, if this maximum accumulative value is smaller than the given threshold which means the occlusion by cloud covering is too heavy, the locating result is meaningless. Of course, the case that the target is beyond the scope of the image is similar to cloud covering. Even if lines are detected from the partially occluded images, it would have little effect on the accumulative amount because the extracted line is too short or the angle varies seriously.

(2)Errors in geometric registration might lead to small rotation angles of images. For such small rotation angles (such as 5°, *i.e*. the angle between two lines is 5°), it will have little influence on the accumulator for $\cos(5^{\circ}) \approx 0.9962$ almost close to 1. Taking the line extraction error into account, this rotation angle can reach 10°. For a 100-pixel-length line feature, the impact on the accumulator of the rotation angle is no more than 2 pixels ($\cos(10^{\circ}) \approx 0.9848$) which is within the tolerance range.

4. Experimental Results and Analysis

4.1. Experimental Results

For experimental validation, we evaluate the proposed target locating technique on screenshots from Google earth. Our proposal is implemented in C++ on a standard i5 Core computer with 2.8 GHz CPU and 4 GB RAM.

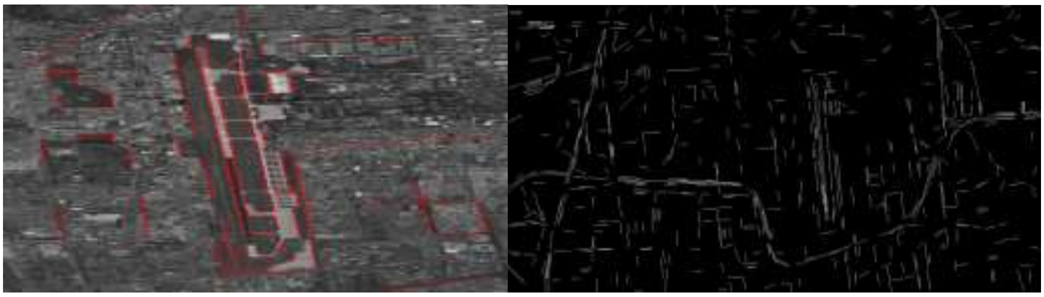
The locating result of Beijing Nanyuan Airport is shown in Figure 5.The size of the input remote sensing image is 1297 × 699. Figure 5(a) represents the line model of the target; Figure 5(b) presents the extracted lines from a remote sensing image containing the airport; Figure 5(c) shows the offset accumulator image of the line model and extracted lines whose grayscale range has been transformed to 0-255.

Figure 5(d) illustrates the corresponding matching results of the maximum position of the offset accumulator figure.

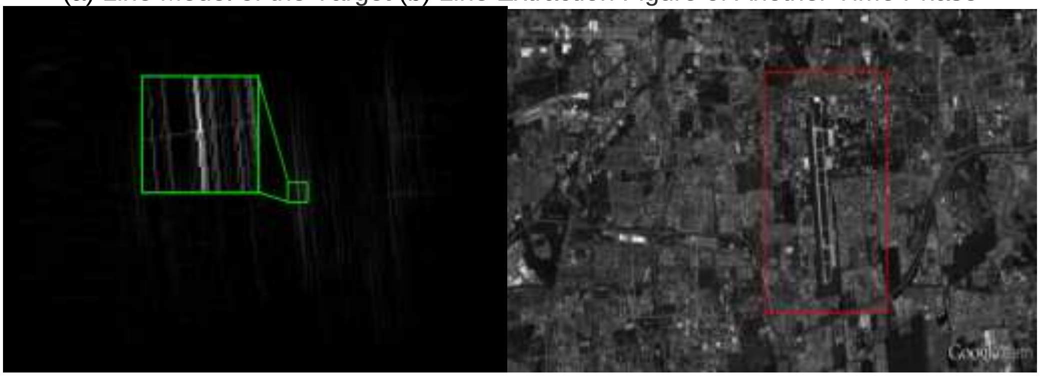
To demonstrate the strong anti-occlusion ability of our target locating technique, 50% of the airport in Figure 6 is occluded by cloud as shown in Figure 6(a). Figure 6(b) presents the offset accumulator;Figure6(c) illustrates the corresponding matching results of the maximum position of the offset accumulator figure. Experimental results show that our target locating technique is quite robust to partial occlusion.

To demonstrate the robustness to small registration error of our target location technique, a small angle of rotation is imposed on Figure 5. Figure 7(a) represents the rotated image; Figure 7(b) presents the offset accumulator; Figure 7(c) illustrates the corresponding matching results of the maximum position of the offset accumulator figure.

Figure 8 shows more locating results on Google earth dataset. The concerned targets include airports and harbors. The corresponding target line models have been omitted for the sake of space.

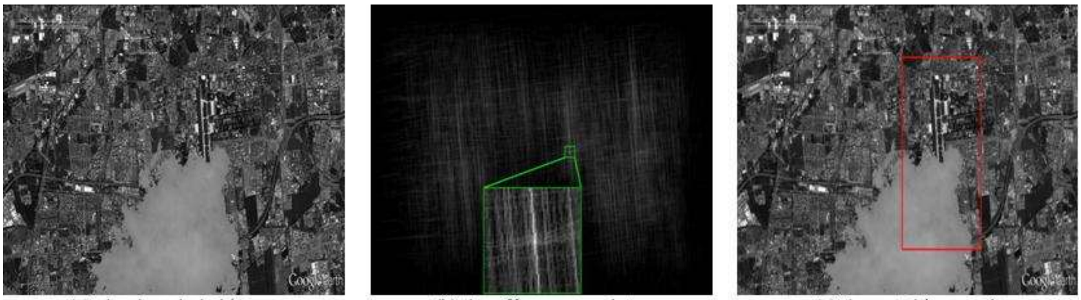


(a) Line Model of the Target (b) Line Extraction Figure of Another Time Phase



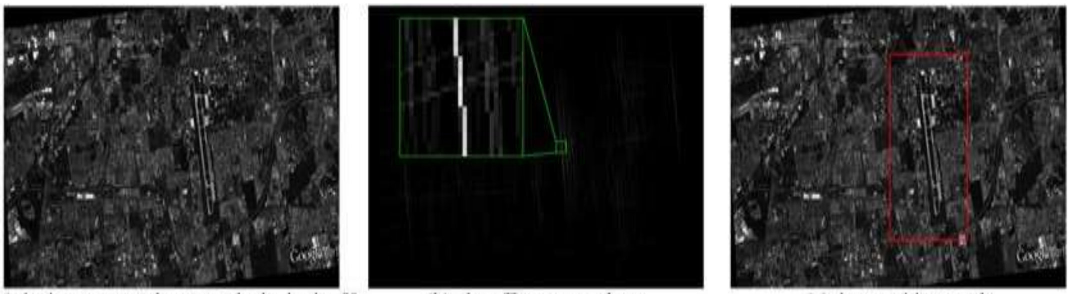
(c) The Offset Accumulator (d) The Matching Result

Figure 5. Locating Process and Results



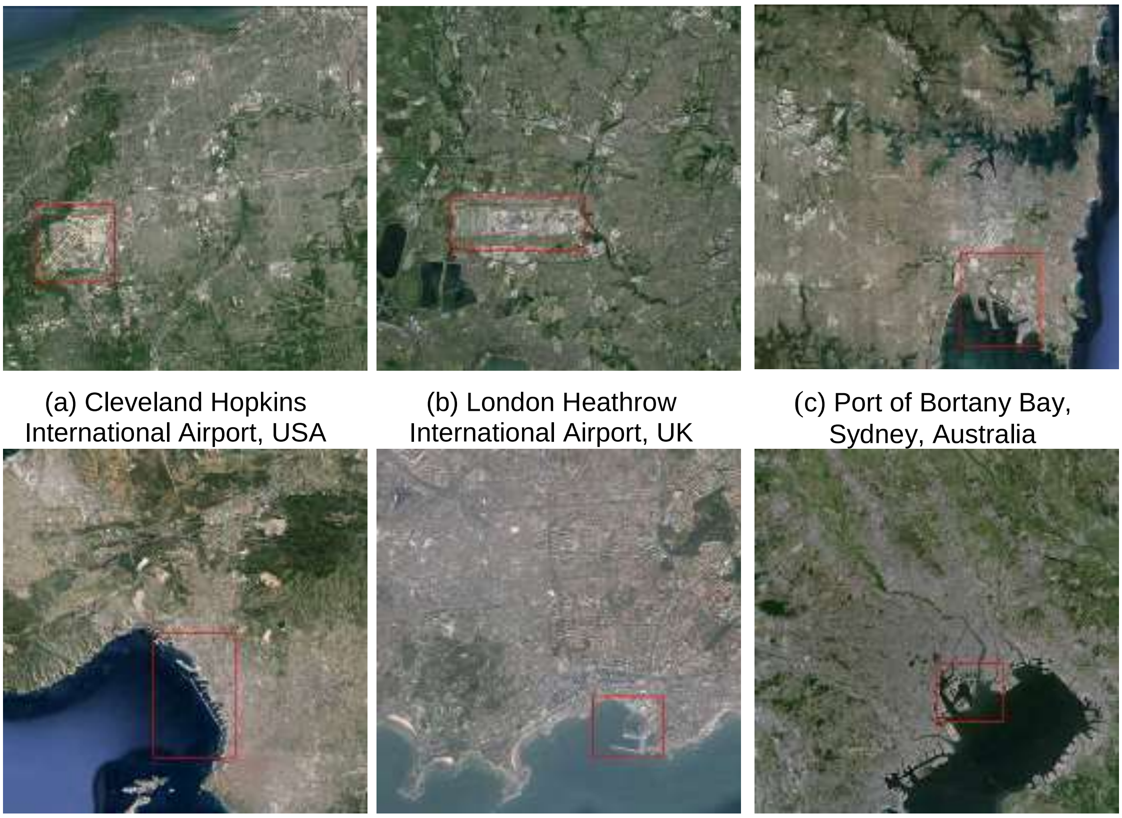
(a) cloud-occluded image (b) the offset accumulator (c) the matching result

Figure 6. Locating Results under Occlusion



(a) the image rotated counterclockwise by 5° (b) the offset accumulator (c) the matching result

Figure 7. Location Results Under the Condition of Small Angle Rotation



(a) Cleveland Hopkins International Airport, USA

(b) London Heathrow International Airport, UK

(c) Port of Bortany Bay, Sydney, Australia

(d) Marseilles Port, France

(e) Qingdao Olympic Sailing Center, China

(f) Tokyo Port, Japan

Figure 8. Some Locating Results on Google Earth Dataset

We also compare our target locating method with the saliency map based algorithm in [16] and the distance transformation based algorithm in [17]. We choose 174 Google earth images in 30m×30m spatial resolution of computation time. The analytical result is as shown in Table 1.

Table 1. Comparison Table of the Three Algorithms on Google Earth Dataset

Algorithms	Recognition rate	Average Time
Saliency map based algorithm	84.5 %	1187 ms
Distance transformation based algorithm	89.1 %	1982 ms
Our algorithm	90.8 %	65 ms

Table 2 gives more detailed information about time consumption and location of the three algorithms. The target to be located is Beijing Nanyuan Airport in Figure 5(a).

Table 2. Comparison Table of Experimental Results

	Location (pixels)				Time consumption(ms)		
	Ground truth	Saliency map based algorithm	Distance transformation based algorithm	Ours	Saliency map based algorithm	Distance transformation based algorithm	Ours
Fig5	(629, 123)	(629, 124)	(627, 122)	(629, 122)	3523	5468	234
Fig6	(629, 123)	(639, 115)	(626, 116)	(625, 125)	3432	5015	219
Fig7	(611, 126)	(654, 140)	(620, 121)	(615, 118)	3871	5392	235

4.2. Analysis of Experimental Results

(1)Precision analysis

The proposed target location technique focuses on the offset accumulations of line pairs. Theoretically, different line pairs contribute to a maximum value in the accumulator. However, due to the influence of the imaging conditions and line extraction algorithms, the true target location is located in the neighborhood of the maximum value. Therefore, the locating accuracy of the proposed method is roughly the neighborhood radius. According to the amplified results in Figure 5 andFigure6, the locating accuracy of the proposed algorithm is at pixel level.

(2)Time consumption

The computational time is consumed mainly in two aspects, line extraction and calculation of accumulator. The computational time consumed for the three figures in Table 2 are 172ms,172msand 171msrespectively, which takes up 70%~75% of the whole computational time. The remaining computational time is consumed on the calculation of the accumulator.

(3) Robustness analysis

In this paper, we validate the robustness of our target location technique against two challenging factors (cloud cover and small angle rotation) and draw the following two conclusions:

1) For cloud covering, our target locating technique can accurately locate the target even when the occlusion ratio reaches over 50%, while the other two compared algorithms suffer from a large shift from the ground truth. Our method achieves strong anti-occlusion ability by calculating the accumulations of the un-occluded lines. Even if some abnormal line segments are extracted from the cloud area, it will not have a significant impaction the locating result.

2) For small rotation angles, the ground truth of the target location will change. The new target location can be calculated by rotating the previous location. As shown in Figure 8 where the image rotates clockwise by 5°, the influence of small rotation on our target locating technique is quite limited.

5. Conclusions

In this paper, a simple and fast target locating technique based on line features is proposed to locate the specific target in a remote sensing image almost in real time. A line model of the target is generated by manual annotation. A robust line extraction algorithm is adopted to extract line segments from the remote sensing image. An offset accumulator between the line model and extracted line segments is calculated. The target location is determined by searching the maximal matching point in the accumulator image.

Experimental results show that our target locating technique can reach pixel level accuracy and nearly real-time computational efficiency. At the same time, this technique can deal with cloud occlusion and small angle rotation conditions. Though this technique cannot achieve sub-pixel locating precision, it can surely be applied to many specific applications as the initial value considering its efficiency and robustness.

Acknowledgments

This work was supported in part by the Foundation of Science & Technology Department of Sichuan Province (Grant No. 2012GZ0088 and No. 2013JQ0005).

References

[1] S. Dawn, V. Saxena and B. Sharme, “Remote sensing image registration techniques: s survey”, International Conference on Image and Signal Processing, Springer Berlin Heidelberg, **(2010)**, pp. 103-112.
[2] B. Zitova and J. Flusser, “Image registration methods: A survey”, Image Vision Computing, vol. 21, no. 11, **(2003)**, pp. 977-1000.
[3] R. Nevatia and K. Price, “Locating structures in aerial images”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 4, no. 5, **(1982)**, pp. 476-484.
[4] X. Yuan and J. Cao, “Theories and Methods on accurate ground target locating for remote sensing images of high resolution”, Beijing: Science Press, **(2012)**, pp.19-20.
[5] L. Ltti, C. Koch and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 11, **(1998)**, pp. 1254-1259.
[6] L. Yu, D. Zhang and E. J. Holden, “A fast and fully automatic registration approach based on point features for multi-source remote-sensing images”, Computers and Geosciences, vol. 34, no. 7, **(2008)**, pp. 838-848.
[7] S. H. Park, K. M. Lee and S. U. Lee, “A line feature matching technique based on an eigenvector approach”, Computer Vision and Image Understanding, vol. 77, no. 3, **(2000)**, pp. 263-283.
[8] H. Li, B. S. Manjunath and S. K. Mitra, “A contour-based approach to multisensor image registration”, IEEE Transactions on Image Processing, vol. 4, no. 3, **(1995)**, pp. 320-334.
[9] S. K. Yong, H. L. Jae and B. R. Jong, “Multi-sensor image registration based on intensity and edge orientation information”, Pattern Recognition, vol. 41, no. 11, **(2008)**, pp. 3356-3365.
[10] L. Yu, S. Xian and H. Wang, “Automatic target detection in high-resolution remote sensing images using a contour-based spatial model”, IEEE Geoscience and Remote Sensing Letters, vol. 9, no. 5, **(2012)**, pp. 886-890.
[11] C. Chen, “Signal and image processing for remote sensing”, CRC Press, vol. 2012, **(2000)**.
[12] Z. Shao and M. Chen, “Line-based matching for high-resolution images with robustness for scale, rotation and illumination”, Optics and Precision Engineering, vol. 21, no 3, **(2013)**, pp. 790-798.
[13] C. Tao, Y. Tan, H. Cai and J. Tian, “Airport detection from large IKONOS images using clustered SIFT keypoints and region information”, IEEE Geoscience and Remote Sensing Letters, vol. 8, no. 1, **(2010)**, pp. 128-132.
[14] A. A. Goshtasby, “2-D and 3-D image registration: for medical, remote sensing”, and industrial applications. John Wiley& Sons, **(2005)**.
[15] G. Wen and R. Wang, “A robust approach to extracting straight lines”, Journal of Software, vol. 12, no. 11, **(2001)**, pp. 1660-1667.
[16] J. Xu, W. Hong and Y. Wu, “A remote sensing image matching algorithm based on distance transformation and genetic algorithms”, Journal of Electronics & Information Technology, vol. 27, no. 7, **(2005)**, pp. 1009-1012.
[17] X. Wang, Q. Lv and B. Wang, “Airport detection in remote sensing images: a method based on saliency map”, Cognitive Neurodynamics, vol. 7, no. 2, **(2013)**, pp. 143-154.

Authors

Xinjian Wang, received the B.S. in computer science from the National University of Defense Technology in 2005.he is currently a PhD candidate at University of Electronic Science and Technology of China(UESTC). His research interest is computer vision.

Guangchun Luo, received the Ph.D. degree in computer science from University of Electronic Science and Technology of China, Cheng du, China, in 2004. He is currently a professor of computer science at the University of Electronic Science and Technology of China, Chengdu, China. His research interests include computer networks, mobile networks, network security and computer vision.

Ke Qin, received his M.E. and PhD degree from the University of Electronic Science and Technology of China in 2006 and 2010 respectively. He was also a visiting scholar at Carleton University in Ottawa, Canada in 2008. His research interests include chaos theory, chaotic neural networks and encryption.

Aiguo Chen, received the Ph.D. degree in computer science from Beijing University of Posts and Tele-communications. His research interests include distributed and cloud computing and information security

Ling Tian, received the Ph.D. degree in computer science from University of Electronic Science and Technology of China. Her research interests include image retrieval, distributed and cloud computing.