Recognition of Airport Runways in FLIR Images Based on Knowledge

Wei Wu, Renbo Xia, Wei Xiang, Bin Hui, Zheng Chang, Yunpeng Liu, and Yahong Zhang

***Abstract*—Airport runway recognition technology would play an important role in developing intelligent weapon systems in the future. In this letter, a method of automatically finding runways in forward looking infrared (FLIR) images is proposed based on the knowledge of vision. First, the line segments in the images are extracted by a fast line segment detector (LSD) and an improved line segment linking method. Then, the regions of interest (ROI) of runways are detected using some constraint rules based on the direction, gradient, and width of line segment pairs. Afterward, an ROI length backtracking technique based on texture distribution is presented to retrieve the complete ROI. Finally, using runway regional self-similarity and contextual information, several decision criteria are formulated to accurately recognize the runway. Experimental results on the FLIR images with different imaging ranges show that the proposed algorithm is robust and has a good real-time performance.**

***Index Terms*—Airport runway recognition, forward looking infrared (FLIR) images, linear feature extraction, region-of-interest (ROI) length backtracking, texture similarity.**

I. Introduction

AIRPORTS have a high strategic value and always get priority attention during the war for air supremacy. Since infrared sensors can work day or night, airport runway recognition based on infrared images instead of visible images has a significant application value and has become a research hot topic in the field of automatic target recognition (ATR).

ATR algorithms [1] for the infrared images are subdivided into three major groups: algorithms based on template matching, algorithms based on machine learning, and algorithms based on knowledge. The first kind of algorithm [2] requires generating accurate reference templates in advance, which is not suitable for recognizing runways consisting only of a pair of parallel line segments. The second kind of algorithm [3] requires plenty of positive and negative sample data acquired in different imaging conditions such as view angle, time, and weather. Both types of algorithms outlined above are unsatisfying in the real applications due to high request for support conditions. In comparison, the third kind of algorithm requires only limited vision knowledge and, therefore, is suitable for recognizing targets with features described easily. As we know that airports have remarkable structural features and gray-level distribution characteristics, it is possible to identify the runways using knowledge. However, there are still some difficulties such as feature description, feature extraction, and decision criterion making. Some related works have been reported in the recent decade. In [4] and [5], the approaches first extract the candidates for runways and recognize them by a trained classifier. In [6], airports are recognized with a new feature named edge segment group. The method is simple and effective but can only deal with infrared images within which runways are located completely. A method based on fuzzy decision [7] is developed using runway shape and regional complexity to identify the airports in different fields of view. Nevertheless, its region-of-interest (ROI) extraction way is not suitable for low signal-to-noise ratio (SNR) or large interference scenarios. In [8], a method is proposed to detect runways in forward looking infrared (FLIR) images by using perspective transformation and runway quadrilateral features, but only adapted to specific shapes of airports. In addition, the multi-information fusion technique is also fully used to detect runways in FLIR image sequences [9].

Since the existing runway recognition algorithms [2]–[9] have poor robustness and cannot adapt to the FLIR images with different imaging distances, a new knowledge-based runway recognition algorithm is developed in this letter. Observing the air-to-ground FLIR images, one can find that the linear structures of airport runways are still noticeable after perspective projection and composed of two approximately parallel long straight edges. Thus, one can detect runways by extracting the parallel line pairs. Our algorithm mainly consists of two parts: ROI extraction and target recognition. First, a fast line segment detector (LSD) [10] is applied to extract the line segments in images. The line segments given by LSD are well localized, but mostly broken. Therefore, an improved line segment linking algorithm is introduced to connect the broken line segments. Then, several constraint rules, such as direction consistency, gradient relativity, and width uniformity, are made to extract parallel line segment pairs as the ROI of runways. For better subsequent recognition, an ROI length backtracking method is proposed to retrieve the complete potential runways in the field of view. In the second part, using runway regional self-similarity and contextual information, several decision criteria are formulated to accomplish the runway recognition efficiently.

II. ROI Extraction

The extraction methods of ROI are usually based on region growing or clustering technologies, but they do not work well

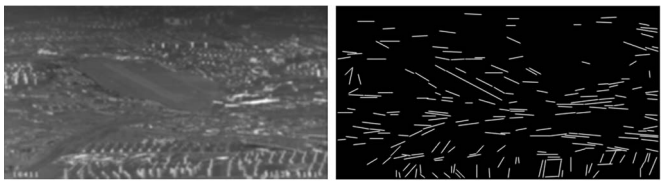
Manuscript received July 8, 2013; revised November 4, 2013 and January 2, 2014; accepted January 6, 2014.

W. Wu and Y. Zhang are with the University of Chinese Academy of Sciences, Beijing 100049, China and also with Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China.

R. Xia, W. Xiang, B. Hui, Z. Chang, and Y. Liu are with the Key Laboratory of Opto-Electronic Information Processing, Chinese Academy of Sciences, Shenyang 110016, China (e-mail: xiarb@sia.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2014.2299898



(a) (b)

Fig. 1. Line segments detection by LSD. (a) Original infrared image with airport. (b) Detected line segments.

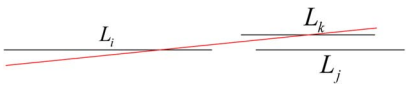


Fig. 2. Line segment linking interference example.

on low SNR infrared images. Another kind of ROI extraction methods based on edges requires a closed target contour. In this letter, an ROI extraction method based on straight line segments is proposed and composed of, e.g., line segment detection, line segment linking, antiparallel line [11] extraction, and ROI length backtracking. The experiments confirm that the proposed ROI extraction method is robust to low SNR infrared images, even in the presence of much clutter.

A. Line Segment Detection

The LSD algorithm [10] is an excellent line segment detection method based on phase grouping and does not require parameter tuning. In [10], a line segment is defined as an image region whose points share roughly the same gradient orientation. The procedure of the LSD algorithm is given as follows: First, pixels that share the same gradient orientation up to a given tolerance are grouped as a connected region, i.e., the line-support region. Then, the line segment that best approximates each line-support region is extracted according to rectangular approximation rules. Finally, each line segment is validated or not based on *a contrario* model. As shown in Fig. 1, LSD still obtains satisfying results even for low SNR infrared images.

B. Line Segment Linking

Due to the low contrast of infrared images or the presence of interferences such as bypass taxiways, the line segments detected by LSD are usually fragmented. In particular, the line segments corresponding to the long linear edges of a runway are generally not continuous [see Fig. 1(b)]. Our purpose is to link these fragmented line segments together to present the runway as a whole. A line segment linking algorithm is proposed in [12]. Given a seed segment L_i and a candidate segment L_k, the algorithm in [12] connects them together if L_k meets the connection rules. The algorithm sometimes makes the new line segment deviate from the true edge because of its lack of optimal judgment. A typically incorrect linking example is shown in Fig. 2. The red line segment in Fig. 2 is the new generated line using the line segment linking method in [12].

We present an improved algorithm for linking the broken line segments. Initially, the line segments generated by LSD are sorted in descending order according to their length. The

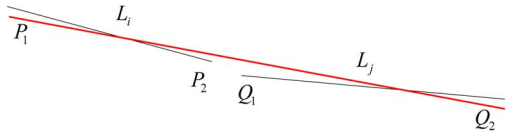


Fig. 3. Line segment linking.

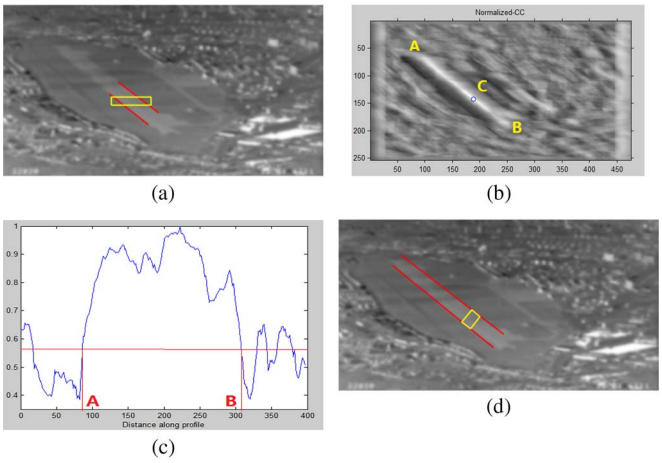
longer segments are preferentially selected as seed segments. Then, given a seed segment as a benchmark, a global search is implemented in the remaining line segments, and all segments that meet the linking rules are considered as candidate linking segments. Finally, the candidate linking segments are scored in (1), and the line segment with the highest score is chosen to be connected with the seed segment. Such a global optimal line judgment method can make L_i connect with L_j in the example in Fig. 2 to get an optimal linking. A least squares algorithm [13] is applied to fit the new line segments in the linking process. The entire process of searching is repeated until all line segments are checked. The adopted line segment linking rules are defined as follows:

1) $\theta(L_i, L_j) < T_{\text{link\_angle}}$;
2) $l_{\min} < T_{\text{gap}}$ and $l_{\max} > l_i + l_j - \Delta l$;
3) $d < T_d$;
4) $|\phi_i - \phi_j| < T_{\text{gd}}$.

As shown in Fig. 3, L_i is the seed segment, and L_j is the candidate segment. θ is the intersection angle between L_i and L_j. $l_{\min}$ and $l_{\max}$ are the minimum and maximum distances between endpoints of the two line segments, respectively. l_i and l_j are the line segment lengths of L_i and L_j, respectively. ϕ_i and ϕ_j are the line gradient orientations of L_i and L_j, respectively, and the line gradient orientation is defined as the average gradient orientation of all pixels on the line segment. d represents the distance of the midpoint of L_j to the line L_i. Rule 1 requires that the angle difference of two line segments is less than a threshold $T_{\text{link\_angle}}$. Rule 2 requires that the gap between two line segments is less than T_{gap}, and Δl is the gap margin. Rule 3 requires that the mass center of the candidate line segment lies in the direction of the seed segment, where T_d is the vertical distance threshold. Rule 4 requires that the two line segments have a similar gradient direction, where T_{gd} is the gradient direction threshold. L_j is stored as a candidate linking segment of L_i if it meets all the aforementioned four rules. The linking cost between L_i and L_j can be defined as follows:

$$s_j = u_1 \frac{T_{\text{gap}} - l_{\min}}{T_{\text{gap}}} + u_2 \frac{T_d - d}{T_d} + (1 - u_1 - u_2) \frac{T_{\text{link\_angle}} - \theta(L_i, L_j)}{T_{\text{link\_angle}}}. \quad (1)$$

When the runway edges are blurred, the directions of the detected line segments corresponding to the runway edges may deviate from the actual runway direction, but the mass centers of the detected line segments are approximately located on the edge of the runway. Based on this characteristic, we assign greater weight to the second coefficient u_2 to improve the robustness of the line segment linking algorithm. In this letter, we use the following values: $u_1 = 0.2$ and $u_2 = 0.5$.



(a) (b) (c) (d)

Fig. 4. Runway length backtracking. (a) Extracted ROI (labeled in red parallel lines). (b) Cross correlogram. (c) Cross-correlation curve. (d) Retrieved complete ROI.

C. Antiparallel Line Extraction

Geometric knowledge of the airport runways indicates that they can be characterized by two parallel line segments with opposite directions, called antiparallel line segments [11]. Based on this consideration, the runway detection can be done by extracting expected antiparallel line segments. The ROI of the runway is defined as the region surrounded by two antiparallel line segments. The antiparallel line segments are extracted by the following several rules on the basis of direction consistency, gradient relativity, and width uniformity.

1) The two line segments are approximately parallel.
2) The gradients of two lines have opposite orientations.
3) The physical width of the two line segments is approximately equal to the actual runway width.

The first rule describes the geometric structure property of a runway. Rule 2 is based on the pixel gray distribution property between the inner region and the two sideward regions of the runway. Rule 3 uses the prior knowledge of the runway width and the physical resolution of real-time images. When severe clutter occurs, the extracted ROI is usually part of the target [see Fig. 4(a)]; thus, a subsequent process is required to retrieve the complete ROI.

D. ROI Length Backtracking

For convenience of recognition, the ROI length should be retrieved as long as possible. According to the regional texture similarity of the runway, an ROI length backtracking method is proposed here as follows.

1) Automatically select a rectangular area through the ROI as a template image, as illustrated in Fig. 4(a). Slide the template in the original image and make a mean residual normalized cross-correlation operation [14] at each pixel position. Equation (2), shown at the bottom of the page, gives a basic definition of the normalized cross-correlation coefficient NC, where $F(x,y)$ and $G(x,y)$ denote the gray level at a pixel in the original image and the template image, and $\overline{F}(x,y)$ is the corresponding mean gray value in the template-sized window centered on the pixel (x,y). $\overline{G}(x,y)$ is the mean gray value of the template image.
2) After the first operation finished, a correlation coefficient matrix is obtained, and the matrix corresponds to a gray-level image [see Fig. 4(b)], called cross correlogram. It is clearly observed that the bright region corresponds to the runway and has a high gray level because of the local self-similarity of the runway. Point C in Fig. 4(b) has the highest gray level because it is the original position of the selected template.
3) Draw a line through the best matching position C in Fig. 4(b) along the runway orientation. The NC values of all points in the line correspond to a curve, named a cross-correlation curve [see Fig. 4(c)]. Because the runway has a high self-similarity, the NC values in the runway are significantly high and continuous. Then, the adaptive threshold algorithm [15] is chosen to extract the middle connected domain between A and B, which correspond to the airport runway endpoints. Fig. 4(d) shows the retrieved runway ROI labeled in red lines.

In practical applications, the template image is just moved along the line through point C and parallel to the runway to improve computational efficiency.

III. Runway Recognition

The target has been limited to a few candidate locations after extracting the ROIs, but there are still some false targets in the candidate locations. To find the real runway, we define two vision knowledge-based constraints: runway context and runway regional self-similarity.

A. Runway Context Information

In most cases, the texture within the runway region is uniformly distributed, but not in both sideward regions. For the FLIR images, the runway texture displays dissimilar visual features when the imaging range varies. The gray distribution of the runway is approximately uniform in the far imaging range because the markings inside the runway are comparatively invisible, but not in the close imaging range. The gray values in both sideward regions of the runway are almost the same

$$\mathrm{NC}(m,n)=\sum_x\sum_y\left(G(x,y)-\overline{G}(x,y)\right)\left(F(x+m,y+n)-\overline{F}(x+m,y+n)\right)\Bigg/\sqrt{\sum_x\sum_y\left(G(x,y)-\overline{G}(x,y)\right)^2\sum_x\sum_y\left(F(x+m,y+n)-\overline{F}(x+m,y+n)\right)^2} \quad (2)$$

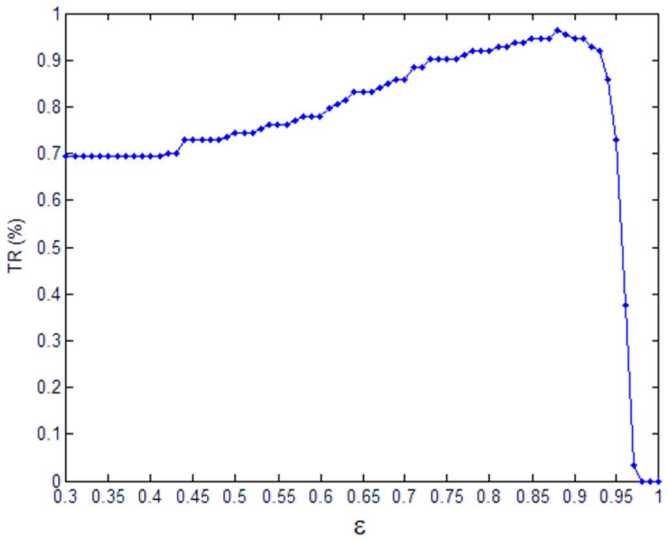


Fig. 5. Parameter analysis for the runway TR rate versus ε.

regardless of the imaging range. Then, the constraint of the gray context of the runway region is defined as follows:

$$\begin{cases} \sigma_1 < 2\sigma_0 \text{ and } \sigma_2 < 2\sigma_0, & \text{far view} \\ \sigma_1 < \sigma_0 \text{ and } \sigma_2 < \sigma_0, & \text{close-up view} \end{cases} \tag{3}$$

where σ_1 and σ_2 are the gray standard deviations of the two sideward regions of the runway. σ_0 is the gray standard deviation of the internal region of the runway. The processed image is considered a far view if its imaging distance is above k km; otherwise, it is a close-up view. Based on the effective identification range for actual infrared imaging guidance, we set k as 6 km in the experiment. A good experimental result is achieved by such a hierarchical constraint method.

B. Runway Regional Self-Similarity Analysis

The significant difference between airport and clutter shows that the texture is approximately uniform in the airport runway, presenting a high regional self-similarity. Based on such feature, a method of airport runway validation is proposed based on the algorithm of mean residual normalized cross correlation [14]. The principle of the proposed method is similar to that presented in Section II-D, but the way of template selection is different. Here, the template image is selected as an inclined rectangular region from the inner area of the candidate runway, as shown in Fig. 4(d), labeled in yellow box. Then, let the template slide within the ROI along the runway and calculate the cross-correlation coefficient in each position. The mean value of all the cross-correlation results is considered as a new measure $\overline{\text{NC}}$. If $\overline{\text{NC}} \geq \varepsilon$, then the ROI will be the true target. This kind of method is rather adaptive to the changes of the image gray value. In our work, ε is set to 0.87, which corresponds to the highest peak in the curve in Fig. 5. The figure depicts the relation between the true recognition (TR) rate of the runway and the parameter ε based on our test data set.

IV. Experimental Results

The recognition performance of the proposed algorithm is tested on several infrared image sequences that were captured

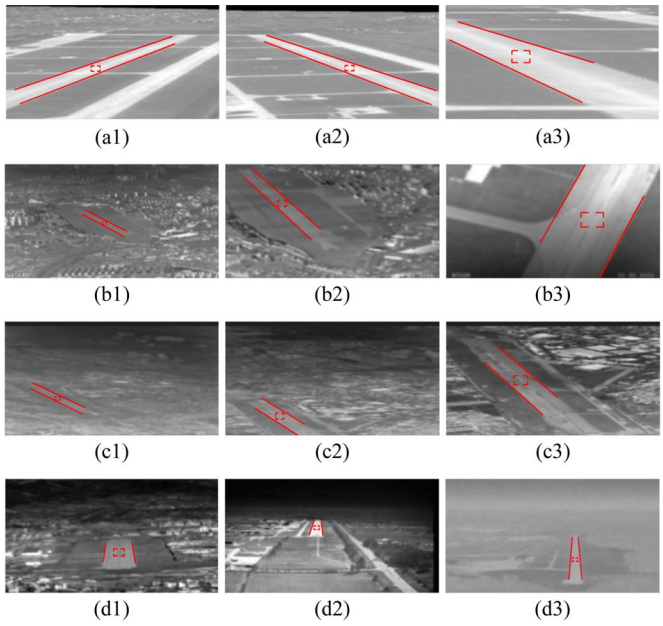


Fig. 6. Runway recognition results with different imaging conditions.

Fig. 7. Runway recognition results in the FLIR images containing no targets.

in different scenes. There are six different airports in our test image sequences, and each sequence contains only one airport. Both single- and double-runway airports are tested in our experiments. In addition, there is a big difference among these test sequences, such as contrast, background, and structural features. The test infrared image has a size of 512 $\times$ 256 and displays a broad variance of each other since they are captured in different imaging ranges, heights, and viewing angles. Some examples are given in Fig. 6, where the recognized runways are labeled in parallel red lines and overlain on the original test images. The centers of dashed red boxes show the aim points. Fig. 6(a) shows the results of runway recognition in the simulated infrared images containing double runways. Only one runway is recognized because the algorithm only returns the longest candidate. It is consistent with the actual single target attack strategy in war. Fig. 6(b) shows the recognition result in real FLIR images under different imaging ranges and viewing angles. Fig. 6(c) gives the recognized runways of another airport. Fig. 6(d) shows the test results of other three different airports. As can be seen, the proposed algorithm successfully locates airport runways and distinguishes runways from roads and taxiways.

In addition, another image sequence containing no airports is also tested. As shown in Fig. 7, there are some false targets in the test images, such as the river, road, and highway bridge, which has a similar structure feature with the airport runway. While no targets are found, the recognition result in Fig. 7 proves that the proposed method has a strong ability to exclude the false targets.

We compare our airport runway recognition algorithm with the algorithm in [9]. Both methods use target knowledge to

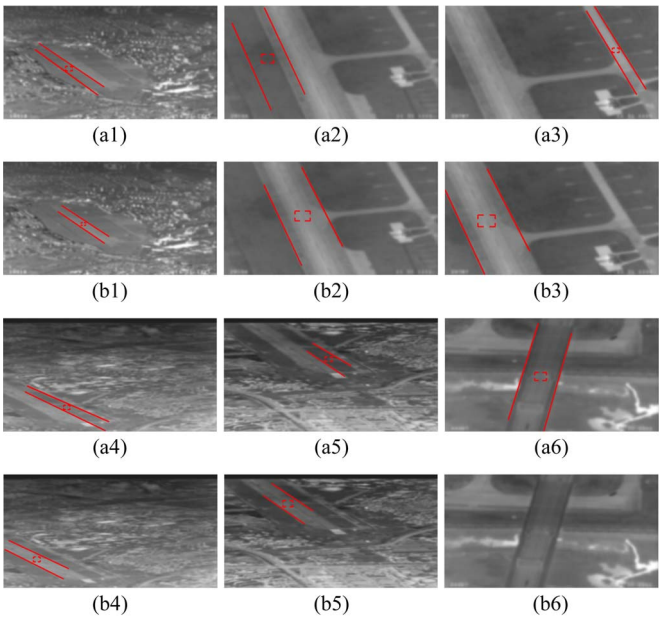


Fig. 8. Runway recognition results comparison. (a) Result of the algorithm mentioned in [9]. (b) Our algorithm's result.

TABLE I
RECOGNITION STATISTICS ON DIFFERENT IMAGE SEQUENCES

Method	Sequence	Total number	Accurate rate /%	False alarm rate /%
[Ours]	IWA	13420	95.98	0
	INA	10593	96.68	3.32
[9]	IWA	13420	83.72	7.65
	INA	10593	85.93	14.07

recognize the runway in infrared images, whereas their implementation strategies are not the same. In [9], the gradient and blob analysis are used to extract the ROI, and the airport runway is recognized by data fusion. A visual comparison between the runway recognition result from the method in [9] and the one from our proposed algorithm is provided in Fig. 8. One can see that the result of runway recognition using [9] is not as good as our method. The method in [9] cannot locate the runway accurately and misrecognizes the main runway with the auxiliary one. In addition, the river bridge [see Fig. 8(a6)] is wrongly judged as the runway by [9].

Table I shows the statistical data of the recognition rate and the false alarm rate of two different methods: ours and the one in [9]. All the tested infrared sequences can be divided into two groups: images with airport (IWA) and images with no airport (INA). It is clear that the proposed algorithm presents a prominent performance on recognizing real targets and excluding false targets, regardless of the imaging condition, backgrounds, and other interference. The recognition algorithm is coded in C language and is run on a PC with CPU 3.06 GHz, RAM 2 GB. The processing time for a single 512×256 FLIR image is a fraction of a second, averaged about 109 ms.

V. CONCLUSION

For FLIR images, a runway recognition algorithm based on target knowledge is proposed in this letter. The new technique integrating LSD and line segment linking has a prominent capacity of extracting line segments of runway edges, even in the presence of low SNR and unpredictable clutter. ROIs are effectively extracted by introducing constraints of direction consistency and gradient relativity. Some knowledge rules are designed based on regional self-similarity and context to enhance the recognition performance of the proposed algorithm. Experimental results prove that the proposed algorithm is robust to different imaging conditions and has a good real-time performance. However, it cannot simultaneously recognize two or more runways in the same scene containing multiple targets. Future work mainly includes two aspects: 1) researching new features of the airport runway to form a more complete set of knowledge and 2) collecting lots of FLIR images under different imaging conditions to test the proposed algorithm.

REFERENCES

[1] B. Bhanu, "Automatic target recognition: State of the art survey," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-22, no. 4, pp. 364–379, Jul. 1986.
[2] Z. R. Zuo, H. F. Chen, and T. X. Zhang, "Study on hit–aim detection of airfield runway based on weighted structure templates matching," in *Proc. SPIE, MIPPR*, 2009, pp. 749532-1–749532-8.
[3] Z. Ugur, H. Ugur, A. Orsan, and U. Ilkay, "Airport runway detection in satellite images by Adaboost learning," in *Proc. SPIE, Image Signal Process. Remote Sens. XV*, 2009, pp. 747708-1–747708-12.
[4] W. Liu, J. W. Tian, and X. W. Chen, "RDA for automatic airport recognition on FLIR images," in *Proc. IEEE WCICA*, 2008, pp. 5966–5969.
[5] W. Liu, J. W. Tian, and X. W. Chen, "PCA based forward-looking infrared airport recognition combining intensity and shape feature," in *Proc. SPIE, MIPPR*, 2007, pp. 67861I-1–67861I-7.
[6] W. Z. Wu, T. X. Zhang, and H. Peng, "Investigation of airdrome runway based on detection of region of interest," *Comput. Meas. Control*, vol. 12, no. 4, pp. 319–321, 2004.
[7] K. M. Yao, L. Q. Song, and J. S. Zhang, "ATR of airport objects in infrared images with complex backgrounds," *Infrared Laser Eng.*, vol. 36, no. 3, pp. 398–402, 2007.
[8] C. H. Ma, J. C. Ye, X. P. Wang, and B. Yang, "Automatic airfield runway recognition in forward looking infrared image," *Infrared Laser Eng.*, vol. 35, no. z4, 2006.
[9] C. Stephan, G. Palubinskas, and R. Müller, "Automatic extraction of runway structures in infrared remote sensing image sequences," in *Proc. SPIE, Image Signal Process. Remote Sens. XI*, 2005, pp. 598208-1–598208-10.
[10] R. G. Von Gioi, J. Jakubowicz, J. M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, Apr. 2010.
[11] A. Huertas, W. Cole, and R. Nevatia, "Detecting runways in complex airport scenes," *Comput. Vis., Graphics, Image Process.*, vol. 51, no. 2, pp. 107–145, Aug. 1990.
[12] G. Q. Lu, H. G. Xu, and Y. B. Li, "Line segment detection based on chain code detection," in *Proc. IEEE ICVES*, 2005, pp. 98–103.
[13] D. York, "Least-squares fitting of a straight line," *Can. J. Phys.*, vol. 44, no. 5, pp. 1079–1086, May 1966.
[14] B. Kai and D. H. Uwe, "Template matching using fast normalized cross correlation," in *Proc. SPIE, Opt. Pattern Recog. XII*, 2001, pp. 95–102.
[15] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002, pp. 600–607.