

# Real-time Accurate Runway Detection based on Airborne Multi-sensors Fusion

Lei Zhang\*, Yue Cheng, and Zhengjun Zhai

*School of Computer Science and Engineering, Northwestern Polytechnical University, Xi'an, 710 072, China*

*\*E-mail: zhangleiactri@163.com*

## ABSTRACT

Existing methods of runway detection are more focused on image processing for remote sensing images based on computer vision techniques. However, these algorithms are too complicated and time-consuming to meet the demand for real-time airborne application. This paper proposes a novel runway detection method based on airborne multi-sensors data fusion which works in a coarse-to-fine hierarchical architecture. At the coarse layer, a vision projection model from world coordinate system to image coordinate system is built by fusing airborne navigation data and forward-looking sensing images, then a runway region of interest (ROI) is extracted from a whole image by the model. Furthermore, EDLines which is a real-time line segments detector is applied to extract straight line segments from ROI at the fine layer, and fragmented line segments generated by EDLines are linked into two long runway lines. Finally, some unique runway features (e.g. vanishing point and runway direction) are used to recognise airport runway. The proposed method is tested on an image dataset provided by a flight simulation system. The experimental results show that the method has advantages in terms of speed, recognition rate and false alarm rate.

**Keywords:** Runway detection; Airborne multi-sensors fusion; Coordinate transformation; EDLines; Line segments linking

## 1. INTRODUCTION

With the rapid development of computer vision technology, multiple existing methods are used to automatically detect meaningful targets from airborne sensing images<sup>1-5</sup>. Simultaneously, some new aviation applications such as enhanced flight vision, vision-aided landing and runway incursion detection have an urgent demand to detect runway target in real-time and recognise it accurately<sup>6-10</sup>. Therefore, airport runway detection from airborne forward-looking images has gained strategic importance in both military and civilian domains. However, airport runway real-time detection still faces with severe challenges. First, runway detection from a large field of view and high-resolution images often brings large time and space complexity. It is necessary to detect runway in real time in order to ensure flight safety. Second, the airports are often located in urban and suburban regions in which riverbank, highway, cornfield and taxiway are morphologically similar to runway. To detect and recognise runway accurately, these distracters should be eliminated from the images thoroughly.

Existing works on airport runway detection can be broadly divided into two categories: feature-based<sup>11-28</sup> and template-based<sup>11,29,30</sup>. The first category relies on the detection of such features as intensity edges, high-contrast corners, or texture primitives. They can be further characterised as line feature-based<sup>11-15</sup>, SIFT feature-based<sup>18-20</sup>, and texture feature-based<sup>21-28</sup>. The line feature-based methods aim to detect the

straight line of runway mainly using Hough transform<sup>11</sup>, radon transform<sup>12</sup>, line segment detector (LSD)<sup>13,14</sup>, and Heuristic line extraction<sup>15</sup>. Although these methods can reach a high speed and a relatively good result, they can be distracted by the edges of rivers, roads, and taxiways and then may cause false alarms easily. In addition, the SIFT feature-based methods apply SIFT features to detect airport runway. Wang<sup>17</sup>, *et al.* extract SIFT feature from regions of candidates and classify them by trained hierarchical discriminant regression (HDR) tree to recognise the airport runway<sup>18</sup>. Tao<sup>19</sup>, *et al.* obtain a set of SIFT key points and use an improved SIFT matching strategy to detect runway. Yao<sup>20</sup>, *et al.* adopt the sparse codes of SIFT as the feature to capture more salient properties of visual patterns instead of raw SIFT features. However, SIFT features still fail to offer sufficiently discriminative power especially for heavily clustered airport regions. Besides, it is time-consuming if the entire blocks of images require to extract SIFT features. Furthermore, the texture feature-based methods focused on characterising, analysing, and interpreting textural features of airport region using machine learning algorithms such as kernel matching pursuits (KMP)<sup>21</sup>, support vector machine (SVM)<sup>22,23</sup>, Adaboost learning<sup>24</sup>, texture model<sup>25,26</sup>, and Pulse coupled neural network<sup>27</sup>. Nevertheless, these algorithms require plenty of positive and negative sample data acquired in different imaging condition such as view angle, time, and weather, they are unsatisfying in the real applications due to high request for support conditions. Moreover, since they are always involved with sliding and overlapped windows, the bottleneck of efficiency will appear. The other category

which is called template-based commonly operates by making point-by-point comparisons between an input image and a pre-stored reference image<sup>28</sup>. They can also be classified into two categories: rigid template-based<sup>29</sup> and deformable template-based<sup>11,30,31</sup>. The former cannot be suitable for all kinds of the airport, however the later cannot be satisfied with real-time requirement due to high time complexity. Although the above algorithms have achieved remarkable progress in runway detection, they neither meet the requirements of real-time and robustness of airborne runway detection nor take full advantage of airborne navigation information.

A novel method for real-time runway detection based on airborne multi-sensor data fusion is proposed. It works in a coarse-to-fine hierarchical architecture. At the coarse layer, a vision projection model from world coordinate system to image coordinate system is built by combining airborne navigation system with image sensor, and then a relatively accurate runway region of interest (ROI) can be extracted. Thus surrounding useless objects and complex background texture can be excluded from ROI so as to reduce false alarms. At the fine layer, EDLines detector is used to extract straight line segments from ROI, and then fragmented line segments generated by EDLines are linked into long line segments based on the direction of runway and angle between two edge lines of runway. Finally, the airport runway is recognised by using vanishing point and angle between two runway edge lines.

## 2. METHODOLOGY

This paper adopts a hierarchical method from coarse to fine to detect runway. Primarily the coarse layer is in charge of runway ROI segmentation based on navigation parameters and navigation database. Through a series of coordinate transformation, an airport runway can be projected onto image plane. Considering the accuracy of navigation parameters, runway ROI based on confidence interval can be estimated. Finally, runway line extraction, linking and recognition are deployed in the fine layer, respectively.

### 2.1 ROI Estimation

To improve runway search efficiency and reduce false alarm rate of runway detection, an estimated runway ROI should be as accurate as possible.

#### 2.1.1 Vision Projection Model

Modern aircrafts are established with inertial navigation instrument (e.g., AHRS, GPS), vision sensors (e.g., FLIR, Visible camera), and navigation database. AHRS can sense the motion state of the aircraft and outputs the aircraft attitude (roll:  $\phi$ , pitch:  $\theta$ , and yaw:  $\psi$ ) in real time. GPS can compute and output geographical coordinates (longitude, latitude, and altitude) of the aircraft. Vision sensors such as FLIR and visible light camera can capture and output forward-looking images of different bands continuously. Navigation database usually includes terrain database and airport database which contains runway details. These above mentioned provide full information to set up a vision projection model. As shown in Fig. 1, It involves four matrix transformations among five coordinate systems.

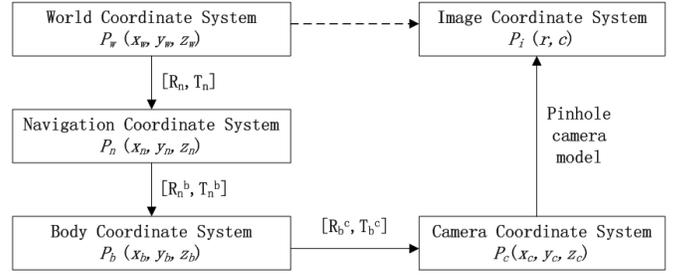


Figure 1. Correlation among different coordinate systems.

Firstly, the point  $P_w$  is mapped to the point  $P_n$  in the navigation coordinate system through matrix transformation  $[R_n, T_n]$ . Secondly, the point  $P_n$  is projected to point  $P_b$  in the body coordinate system through matrix transformation  $[R_n^b, T_n^b]$ . Thirdly, the point  $P_b$  is transformed to the point  $P_c$  ( $x_c, y_c, z_c$ ) in the camera coordinate system through matrix  $[R_b^c, T_b^c]$ . Finally, the point  $P_c$  is projected to the pixel  $P_i$  ( $r, c$ ) in the image coordinate system by the principle of pinhole camera model<sup>32</sup>, as shown in Fig. 2.

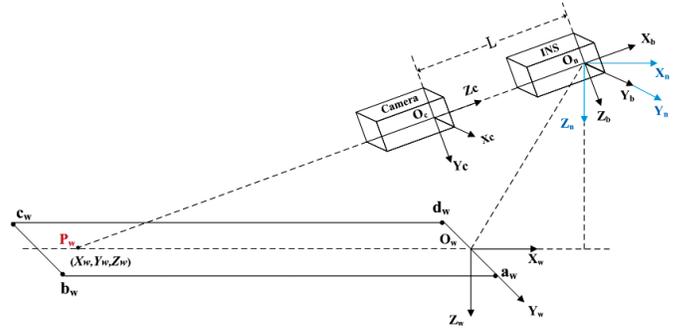


Figure 2. Projection model during approach.

Thus, mathematical model of vision projection can be established as follow.

$$r = \frac{f_v}{\rho} \left[ (x_w - x_a)(\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) + (y_w - y_a) (\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) + (z_w - z_a)(\cos \theta \cos \phi) \right] \quad (1)$$

$$c = \frac{f_u}{\rho} \left[ (x_w - x_a)(\cos \psi \sin \theta \sin \phi - \sin \phi \cos \theta) + (y_w - y_a) (\cos \psi \cos \phi + \sin \psi \sin \theta \sin \phi) + (z_w - z_a)(\cos \theta \sin \phi) \right] \quad (2)$$

where  $\rho$  is the normalisation coefficient,  $f_u = \alpha f$ ,  $f_v = \beta f$ .  $\alpha$  and  $\beta$  are the scale factors in image  $u$  and  $v$  axes, and  $f$  is the focal length of the pinhole camera.

#### 2.1.2 Runway Region Estimation

Due these errors of camera parameters and navigation parameters, the projection of runway zone in the image coordinate system is inaccurate by using projection model simply. The error of navigation parameters has more prominent influence on the accuracy of runway projection region. Especially, the aircraft attitude errors have larger impact than position errors on pixel projection accuracy and belong to primary factors. Therefore, the errors transfer equations of vision projection model can be given as follow:

$$\Delta r = \frac{\partial r}{\partial x_a} \Delta x_a + \frac{\partial r}{\partial y_a} \Delta y_a + \frac{\partial r}{\partial z_a} \Delta z_a + \frac{\partial r}{\partial \phi} \Delta \phi + \frac{\partial r}{\partial \theta} \Delta \theta + \frac{\partial r}{\partial \psi} \Delta \psi \quad (3)$$

$$\Delta c = \frac{\partial c}{\partial x_a} \Delta x_a + \frac{\partial c}{\partial y_a} \Delta y_a + \frac{\partial c}{\partial z_a} \Delta z_a + \frac{\partial c}{\partial \phi} \Delta \phi + \frac{\partial c}{\partial \theta} \Delta \theta + \frac{\partial c}{\partial \psi} \Delta \psi \quad (4)$$

where  $\Delta x_a$ ,  $\Delta y_a$ , and  $\Delta z_a$  are position errors,  $\Delta \phi$ ,  $\Delta \theta$ , and  $\Delta \psi$  are attitude errors.  $\Delta r$  is the error of pixel row and  $\Delta c$  is the error of pixel column.

Though simulating a real landing procedure under different parameters noise level, the projection zone of runway can be estimated and the ratio of the pixel numbers in ROI to the pixel numbers in CCD can be computed. As shown in Fig. 3, the projection zone of runway in the image coordinate system is a quadrangle. At the 100 feet above runway elevation the projection region of runway becomes larger gradually with the noise level of position-attitude parameters increasing.

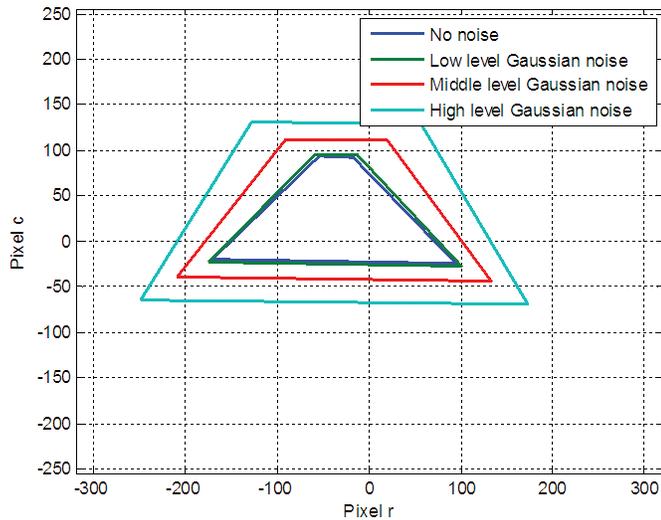


Figure 3. Effects of Gaussian noise on the projection zone.

The above simulation results show that when equipped with high precision navigation device, the ratio of pixels in quadrangle to total pixels in CCD is less than 10 per cent. Obviously, this method is faster than others which process the whole image. Even though using low precision navigation unit, the pixel ratio is less than 25 per cent. To sum up, image processing efficiency is significantly improved by using runway region estimation.

## 2.2 Runway Detection and Recognition

To detect and recognise runway lines from ROI accurately, an real-time line segments detection and a robust runway recognition are essential and indispensable.

### 2.2.1 Line Segments Detection

Due to its special geometry, line segments give a high-level description of airport runway. An ideal line segment detection algorithm could process any images regardless of its origin, orientation or size, and produce robust and accurate line segments in real-time without parameters tuning. Among existing algorithms, line segments detector (LSD)<sup>33-35</sup> and EDLines detector<sup>36-38</sup> are able to satisfy the above-mentioned

requirements. However, EDLines runs up to 11 times faster than LSD<sup>36</sup>, which makes it more suitable for real-time runway detection. As shown in Fig. 4, line segments are extracted from the same two images through EDLines Detector and LSD respectively.

EDLines detector uses edge drawing algorithm<sup>39,40</sup> to detect image and produce a set of clean, contiguous chains of pixel which intuitively correspond to object boundaries. Sobel operator is applied to each pixel and to obtain gradient values  $G_x$  and  $G_y$ . If  $G_x + G_y$  is bigger than a given threshold. If  $G_x > G_y$ , then a vertical edge is pass through this pixel; Otherwise a horizontal edge is assumed to pass through. Then, line segments are extracted from the generated chains of pixel by the Least Squares Line Fitting Method. Finally, the Helmholtz principle is used to eliminate false line segment detections. Let A be a segment of length  $n$  with at least  $k$  points having their direction align with the direction of A in an image of size  $N \times N$  pixels. 'number of false alarms (NFA)' of a line segment is defined as:

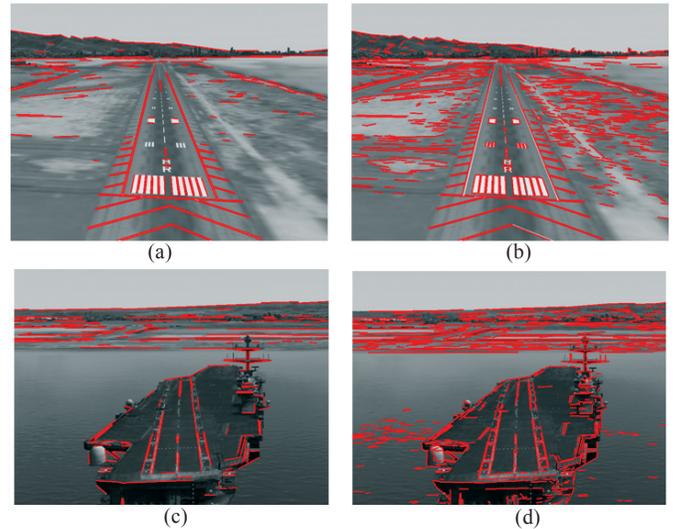


Figure 4. Runway lines detection results from EDLines and LSD: (a) EDLines: 180 lines, 163 ms, (b) LSD: 818 lines, 678 ms, (c) EDLines: 198 lines, 179 ms, and (d) LSD: 755 lines, 626 ms.

$$NFA(n, k) = N^4 \cdot \sum_{i=k}^n p^i \cdot (1-p)^{n-i} \quad (5)$$

If  $NFA(n, k) \leq \epsilon$ , it is a valid line segment. EDLines not only extract perfectly line segments, but it also achieves this in blazing speed compared to other line detectors<sup>41</sup>.

### 2.2.2 Line Segments Linking

Due to low illumination, weak contrast, blurry, occlusion, or clutter in the original image, there are three major problems when the current line segments detectors (e.g. Hough, LSD, and EDLines) are applied to practical images: (1) the detected object edges are composed of small line segments with different orientations, (2) there are often missing line segments (gap-filling segments), leading to discontinuity of edges of the infrastructure objects, and (3) all endpoints of object edges are successfully detected<sup>42</sup>. Our purpose is to link the fragmented line segments together to present the runway as a whole.

As shown in Fig. 5, straight lines  $L_{ab}$  and  $L_{cd}$  (red) are ideal runway lines calculated by vision projection model based on runway prior knowledge. There are many small line segments generated by EDLines in each neighbourhood of the ideal runway lines.

When the aircraft is still far from the runway, the projection region of runway onto the image plane is relatively small. So the neighbourhood of the left runway line  $L_{ab}$  and the neighbourhood of the right runway line  $L_{cd}$  will partially overlap. Some of the line segments will fall into two neighbourhoods simultaneously. It is necessary to distinguish these line segments which belong to its neighbourhood. Then, line segments in each neighbourhood are linked together individually to present a complete runway.

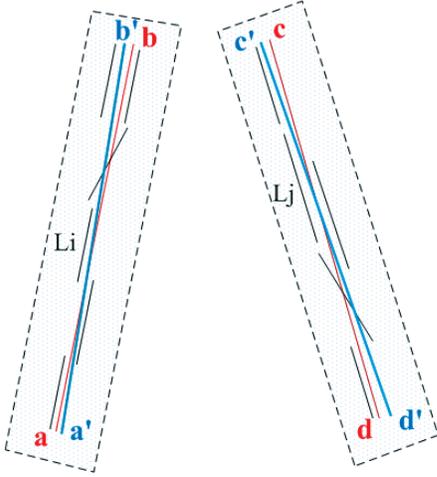


Figure 5. Runway line segments linking.

This paper proposes a fast method to classify and fit two sets of adjacent line segments into two long runway lines. Firstly, some formal symbols should be defined to depict these line segments quantitatively.  $d(L_i, L_{ab})$  or  $d(L_i, L_{cd})$  is the distance from the center of straight line segment to the ideal runway line  $L_{ab}$  or  $L_{cd}$ .  $T_d$  is the width of runway line neighbourhood.  $\theta(L_i, L_{ab})$  or  $\theta(L_i, L_{cd})$  is the angle between any line segment  $L_i$  and the ideal runway line  $L_{ab}$  or  $L_{cd}$ .  $T_\theta$  is the threshold of angle between any line segment  $L_i$  and the ideal runway line  $L_{ab}$  or  $L_{cd}$ . Secondly, if  $d(L_i, L_{ab}) \leq T_d$ ,  $\theta(L_i, L_{ab}) \leq T_\theta$ , and  $d(L_i, L_{ab}) \leq d(L_i, L_{cd})$  then the line segment  $L_i$  belongs to the neighbourhood of line  $L_{ab}$ . If  $d(L_i, L_{cd}) \leq T_d$ ,  $\theta(L_i, L_{cd}) \leq T_\theta$ , and  $d(L_i, L_{cd}) \leq d(L_i, L_{ab})$  then the line segment  $L_i$  belongs to the neighbourhood of line  $L_{cd}$ . Therefore, these line segments are divided into two sets ( $S_L$  and  $S_R$ ). Thirdly, fragmented line segments in each set are condensed into a long runway line by the Least Squares Line Fitting Method. As shown in Fig. 5, straight line segments ( $L_1, \dots, L_i, \dots, L_k$ ) in the neighbourhood of line  $L_{ab}$  are linked into the left runway line  $L_{a'b'}$  (in blue). Meanwhile, straight line segments ( $L_1, \dots, L_j, \dots, L_m$ ) in the neighbourhood of line  $L_{cd}$  are connected into the right runway line  $L_{c'd'}$  (in blue).

### 2.2.3 Features Recognition

Although the target has been detected after extracting the ROI, but there are still some false targets which should

be excluded from runway lines candidates. So some stable features should be used to recognise runway accurately. They can be easily obtained from the results of image detection and be insensitive to weather and surroundings, such as the runway direction, length, the angle between two edge lines, and the vanishing point. Finally, these runway features detected from image should be compared with ideal values estimated by vision projection model. If the detected values from the image are coincide with the estimates, then the runway can be identified. These criterions to identify the airport runway are formalised as follow:

The slop of left or right runway edge line:

$$slop_{left} = \frac{a_r - b_r}{a_c - b_c} \quad (6)$$

$$slop_{right} = \frac{d_r - c_r}{d_c - c_c} \quad (7)$$

The length of left or right runway edge line:

$$Length_{left} = \sqrt{(a_r - b_r)^2 + (a_c - b_c)^2} \quad (8)$$

$$Length_{right} = \sqrt{(c_r - d_r)^2 + (c_c - d_c)^2} \quad (9)$$

The angle between two edge lines:

$$Angle = \left| \arctan \left( \frac{slop_{left} - slop_{right}}{1 + slop_{left} * slop_{right}} \right) \right| \quad (10)$$

The vanishing point coordinate of runway parallel lines:

$$VPoint_{row} = \frac{(c_c - a_c)(b_r - a_r)(d_r - c_r)}{(b_c - a_c)(d_r - c_r) - (d_c - c_c)(b_r - a_r)} + \frac{a_r(b_c - a_c)(d_r - c_r) - c_r(d_r - c_r)(b_r - a_r)}{(b_c - a_c)(d_r - c_r) - (d_c - c_c)(b_r - a_r)} \quad (11)$$

$$VPoint_{column} = a_c + \left( \frac{b_c - a_c}{b_r - a_r} \right) * Vpoint_{row} - a_r \left( \frac{b_c - a_c}{b_r - a_r} \right) \quad (12)$$

where  $(a_r, a_c)$ ,  $(b_r, b_c)$ ,  $(c_r, c_c)$ , and  $(d_r, d_c)$  are the pixel coordinates of runway four vertices ( $a_w, b_w, c_w$ , and  $d_w$ ), which can be computed individually by the vision projection model. Especially as an intersection point of two runway lines at infinity, the vanishing point should fall into a specific small area in image coordinate system based on corresponding probability. It is an ideal feature with stable and accurate characteristic to recognise runway lines in practice. Through the above theoretical computation in section 2.1.2, the approximate location of a vanishing point with corresponding probability can be estimated.

### 3. ALGORITHM AND COMPLEXITY

The complete algorithm in pseudo-code is as the follow:

Inputs: an image *Imag*, these coordinates of runway four vertices ( $a_w, b_w, c_w$ , and  $d_w$ ) in the world coordinate system, and navigation parameters ( $x_a, y_a, z_a, \phi, \theta$ , and  $\psi$ ).

Outputs: runway detected in image *Imag*.

1  $ROI \leftarrow EstimateROI(x_a, y_a, z_a, \phi, \theta, \psi)$ ; // Apply project

model to estimate runway region of candidate

- 2  $Imag = AdaptHistogramEquation(ROI)$ ; // Apply Adaptive Histogram Equation on ROI to enhance image
- 3  $V = EDLines(Imag)$ ; // Apply EDLines Detector on image  $Imag$  to get a set of line segments  $V$
- 4  $Left \leftarrow \phi$ ; // Initialise the set of runway left edge line segments
- 5  $Right \leftarrow \phi$ ; // Initialise the set of runway right edge line segments
- 6 for each line segment  $L_i \in V$  // Select line segments which are adjacent to  $L_{ab}$  or  $L_{cd}$
- 7 if  $d(L_i, L_{ab}) \leq T_d$  and  $\theta(L_i, L_{ab}) \leq T_\theta$  and  $d(L_i, L_{cd}) \leq d(L_i, L_{ab})$  then
- 8  $Left \leftarrow Left \cup L_i$ ;
- 9 else if  $d(L_i, L_{cd}) \leq T_d$  and  $\theta(L_i, L_{cd}) \leq T_\theta$  and  $d(L_i, L_{cd}) \leq d(L_i, L_{ab})$  then
- 10  $Right \leftarrow Right \cup L_i$ ;
- 11 end
- 12 end
- 13  $Runway\_Left\_Line = LineFit(Left)$ ; // To fit runway left edge line by the Least Squares Method
- 14  $Runway\_Right\_Line = LineFit(Right)$ ; // To fit runway right edge line by the Least Squares Method
- 15  $\theta_{LR} = Angle(Runway\_Left\_Line, Runway\_Right\_Line)$ ; // Compute the angle between left edge line and right edge line
- 16  $(Vp_{row}, Vp_{column}) = VPoint(Runway\_Left\_Line, Runway\_Right\_Line)$ ; // Compute the vanishing point pixel coordinate of two runway edge lines
- 17 if  $\min \theta_{LR} \leq \theta_{LR} \leq \max \theta_{LR}$  and  $(Vp_{row}, Vp_{column}) \in VP_{zone}$  then
- 18  $draw(Imag, Runway\_Left\_Line, Runway\_Right\_Line)$ ; // Mark runway in image  $Imag$
- 19 end

This algorithm has four parts: extraction of ROI, line segment detection, line segment linking, and runway lines recognition. Extraction of ROI has an  $O(1)$  time complexity and an  $O(n)$  space complexity where  $n$  is total pixel number in the ROI. The image enhancement using adaptive histogram equalisation has an  $O(n)$  time complexity. Meanwhile, the runtime complexity of EDLines reveals  $O(n)$  as a linear algorithm. Because many line segments outside the neighbourhood of ideal runway lines are excluded, the number of remaining line segments that need to be linked into long lines is less. To link line segments is not very time consuming, the complexity is  $O(m + p)$ ,  $m$  is the number of fragmented line segments extracted from the ROI,  $p$  is the number of line segments which is collinear with line  $L_{ab}$  or  $L_{cd}$ . The last step of runway lines recognition consumes time  $O(q^2)$ ,  $q$  is a number of candidate runway lines. To sum up, the whole process has a runtime complexity of  $O(n+m+p+q^2)$ .

#### 4. EXPERIMENTS AND RESULTS

All experiments are performed on a computer with Intel Core i7-5500U processor clocked at 2.40GHz and 8.0 GB memory. All codes are run in Matlab 2013a.

#### 4.1 Experimental Condition

In order to verify the proposed method, the authors take advantage of VT MAK virtual reality simulation development software (VT-MAK simulator) to generate visual and infrared video corresponding with airborne navigation parameters. This high-fidelity video provides a realistic view of out-of-cabin environment with different weather conditions, time of day, atmosphere conditions, sky textures. The focus length of camera is 36 mm, and the pixel size of CCD is 17  $\mu$ m. The resolution of forward-looking image is 600 $\times$ 800 pixels. The accuracy of position parameters follows:  $\Delta x_a \sim N(0, 0.1^2)$ ,  $\Delta y_a \sim N(0, 0.1^2)$ , and  $\Delta z_a \sim N(0, 0.1^2)$ . The accuracy of attitude parameters follows:  $\Delta \phi \sim N(0, 0.01^2)$ ,  $\Delta \theta \sim N(0, 0.01^2)$ , and  $\Delta \psi \sim N(0, 0.02^2)$ . In addition, the airport runway is 3,000 m long and 60 m wide, and the carrier deck is 326 m long and 76 m wide. The aircraft approaches and descends along the glide path above runway smoothly.

#### 4.2 Experimental Results

In Fig. 6, this paper presents some detection results of our proposed method. Both visible image and infrared image are all applicable to the proposed method, and better results are obtained. Scenario-1 infrared images (IR) in row 1 are obtained above the airport with latitude (Lat) = 21.306869 $^\circ$ N, longitude (Lon) = -157.970263 $^\circ$ E, altitude (Alt) = 771m,  $\phi = -109^\circ$ ,  $\theta = 7.8^\circ$ , and  $\psi = -49^\circ$ . Scenario-2 visual images (Vis) in row 2 are obtained with Lat= 21.306897 $^\circ$ N, Lon= -157.948612 $^\circ$ E, Alt=93m,  $\phi = -112.7^\circ$ ,  $\theta = -4.8^\circ$ , and  $\psi = -55.1^\circ$ . Scenario-3 visual images in row 3 are obtained with Lat= 21.299724 $^\circ$ N, Lon= -157.952523 $^\circ$ E, Alt=384m,  $\phi = -119.4^\circ$ ,  $\theta = -21.5^\circ$ , and  $\psi = -43.6^\circ$ . Scenario-4 infrared images in row 4 are obtained above the aircraft carrier with Lat= 21.290395 $^\circ$ N, Lon= -157.935106 $^\circ$ E, Alt=167m,  $\phi = 159.0^\circ$ ,  $\theta = -38.5^\circ$ , and  $\psi = 42^\circ$ . Scenario-5 visual images in row 5 are obtained with Lat=21.294766 $^\circ$ N, Lon=-157.940865 $^\circ$ E, Alt=77m,  $\phi = -175.8^\circ$ ,  $\theta = -68.2^\circ$ , and  $\psi = 17.6^\circ$ . Scenario-6 visual images in row 4 are obtained with Lat= 21.300316 $^\circ$ N, Lon= -157.940726 $^\circ$ E, Alt=107m,  $\phi = 133.9^\circ$ ,  $\theta = 74.2^\circ$ , and  $\psi = 152.5^\circ$ . At the coarse layer of our model, the ROI is highlighted in blue at the left column with the detected line segments in red. At the fine layer of our model, the detected line segments are linked into two long lines which are shown in red at the right column.

The proposed method is applied to different approach scenarios, and the experimental results are shown in Table 1. The experimental result shows that the proposed algorithm is real time to detect runway target and robust to various approach scenarios.

#### 4.3 Parameter Selection

To statically analyse the effect of different parameter values on runway detection efficiency, airport images under different approach scenarios are tested. In this experiment, each parameter value is adjusted so as to achieve the best detection effect. For  $T_d$ , the optimal parameter is around 7 pixels, and the detection performance is generally robust when  $5 \leq T_d \leq 10$ . If this value is too large, the distance between estimated vanishing point and detected vanishing point will

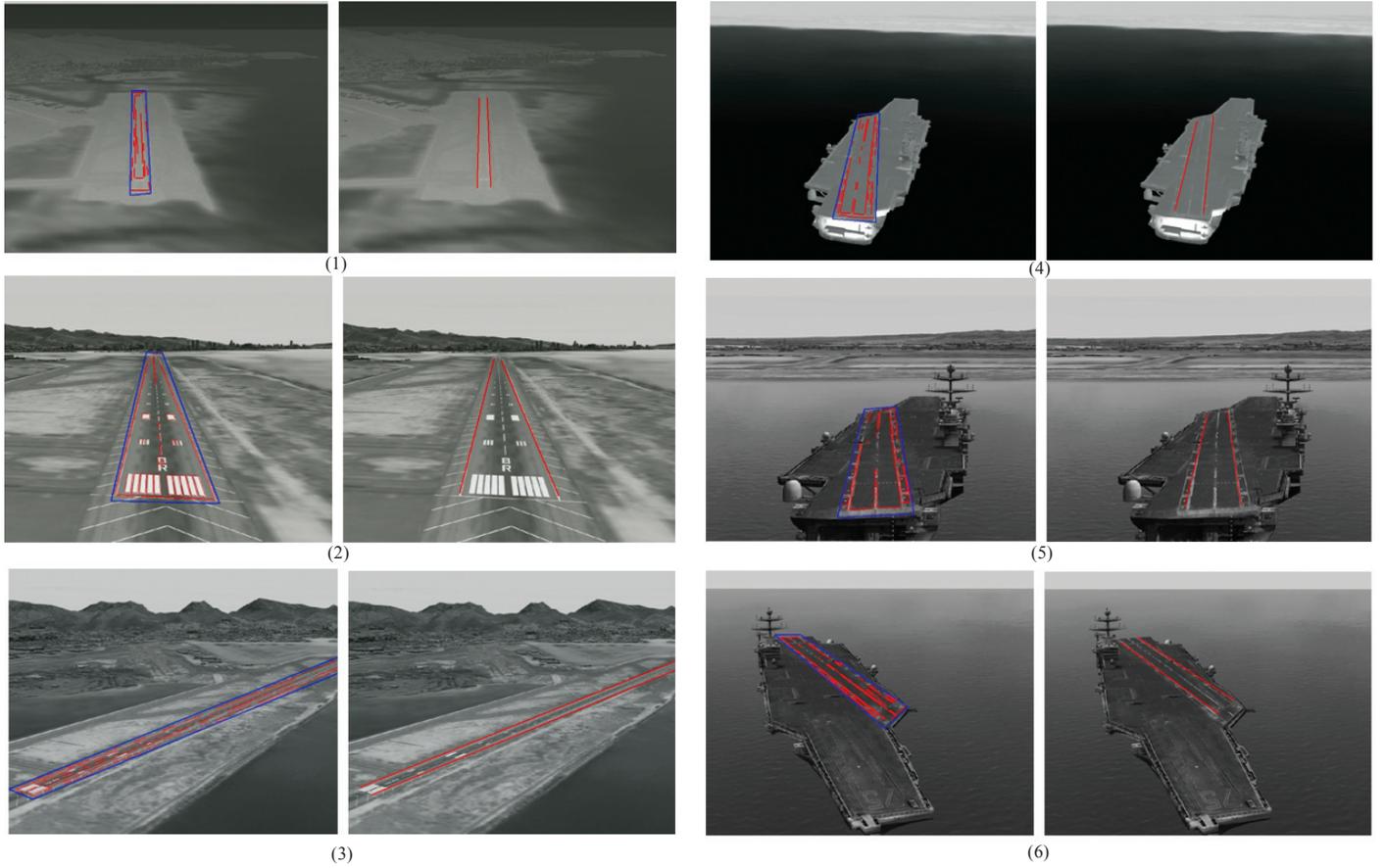


Figure 6. Airport runway detection results under different position and attitude.

Table 1. Experimental results under different Scenarios

Scenarios	Vis/IR	ROI (pixels)	ROI/CCD ratio	Lines	Time (ms)	$Vp \ \delta r$	$Vp \ \delta c$	Estimated $Vp (r,c)$	Detected $Vp (i)$
1	IR	3005	0.0063	23	62	0.6771	0.5633	340.4103 -45.2869	339.4243 -46.2446
2	Vis	70794	0.1475	84	228	0.6181	0.5303	367.9412 132.8195	368.9302 132.2792
3	Vis	4769	0.0099	128	349	0.6053	0.5453	3423.7053 -876.6417	3422.6 -877.6863
4	IR	7406	0.0154	45	44	0.6109	0.5994	438.1768 -15.8315	437.3359 -15.3316
5	Vis	20667	0.0431	54	123	0.6244	0.6201	428.4943 160.7787	427.9946 161.4777
6	Vis	4493	0.0094	53	144	0.6333	0.5591	-299.1825 -223.5697	-298.5402 -222.5707

become longer. Then the detected vanishing point may fall out of its expected region. Furthermore, the runway cannot be recognised correctly. For  $T_0$ , the detection performance is ideal when  $-2^\circ \leq T_0 \leq 2^\circ$ . when  $|T_0|$  is larger than  $3^\circ$ , the runway detection rate declines dramatically.

The runway detection rate also increases dramatically with the projection region ( $r - \Delta r \leq \hat{r} \leq r + \Delta r$ ,  $c - \Delta c \leq \hat{c} \leq c + \Delta c$ ) of vanishing point rising. Obviously, the selection of projection errors ( $\Delta r$ ,  $\Delta c$ ) of vanishing point will affect detection result directly. In Scenario-2, Monte Carlo simulation is run over each of four scenarios to analyse the probability distribution of pixel errors of vanishing point. As shown in Fig.7,

statistic histograms of  $\Delta r$  and  $\Delta c$  represent their probability distribution functions (pdf). Both the pdf of  $\Delta r$  and the pdf of  $\Delta c$  obey Gaussian with low level noise. Statistic data can also be achieved:  $\delta r = 0.6181$ ,  $\overline{\Delta r} = 5.4519e-04$ ,  $\delta c = 0.5303$ ,  $\overline{\Delta c} = -0.0088$ ,  $P(-2 * \delta r \leq \Delta r \leq 2 * \delta r) = 0.9443$ , and  $P(-2 * \delta c \leq \Delta c \leq 2 * \delta c) = 0.9381$ . The probability of vanishing point appeared in estimated area is 0.886 when  $-2 * \delta r \leq \Delta r \leq 2 * \delta r$  and  $-2 * \delta c \leq \Delta c \leq 2 * \delta c$ .

#### 4.4 Comparison with other Algorithm

This paper compares the proposed method with the two

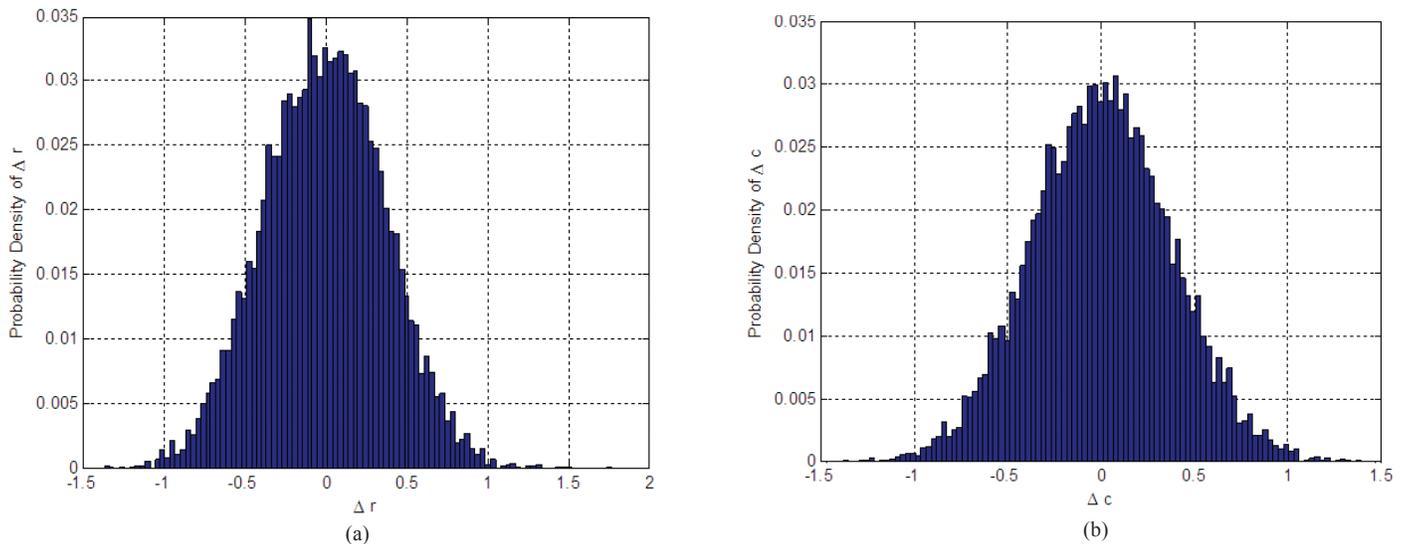


Figure 7. Probability distribution of vanishing: (a) Point row error and (b) Point column error.

methods<sup>11</sup>. The reason why these two algorithms are chosen for comparison is that they follow a similar coarse-to-fine framework to the proposed method. Different from the proposed method, they use horizon to extract search region and then recognise the airport runway individually by edge detection and template matching.

The search space for runway detection<sup>11</sup> is only restricted to below the horizon. However, the runway detection region in the proposed method is relatively accurate, so it can save more time. Comparison with other algorithms under scenario 1 is as shown in Table 2.

Table 2. Comparison with other algorithms under scenario 1

Models	Detected	Not detected	False detected	Ave. time (ms)
Edge detection	73	88	39	421.2
Template matching	114	56	48	627.3
Proposed	131	21	48	158.3

The experimental results in Table 2 indicate that the proposed method has advantages in both the real-time and the robustness, which takes full advantages of the airborne navigation information to estimate runway ROI precisely. So, the search region of runway will shrink dramatically. In addition, the fastest line detector, EDLines, can also extract line segments in real time. Meanwhile, vanishing point feature is adopted to strengthen the robustness of runway recognition.

## 5. CONCLUSIONS

In this paper, a real-time and accurate runway detection method based on airborne multi-sensors data fusion is presented. The model consists of two main steps: coarse runway region of candidate and fine runway detection. Comprehensive evaluation on test set and comparisons with two state-of-art algorithms have demonstrated the effectiveness and efficiency of the proposed work.

In future works, the authors will use more types of image, i.e. forward-looking infrared image and radar image, to evaluate the proposed method. Also, this algorithm will be run

on the embedded computer to test its efficiency. Furthermore, the proposed method will be applied to real-time airborne applications such as vision-aided navigation and enhanced flight vision.

## REFERENCES

- Lai, J.; Mejias, L. & Ford, J.J. Airborne vision-based collision-detection system. *J. Field Robot.*, 2011, **28**(2), 137-157.  
doi: 10.1002/rob.20359
- Bar, D.E.; Wolowelsky, K.; Swirski, Y.; Figov, Z.; Michaeli, A.; Vaynzof, Y.; Abramovitz, Y.; Ben-Dov, A. & Yaron, O. Target detection and verification via airborne hyper spectral and high-resolution imagery processing and fusion. *IEEE Sen. J.*, 2010, **10**(3), 707-711.  
doi: 10.1109/JSEN.2009.2038664
- Liu, C.J.; Zhang, Y.; Tan, K.K. & Yang, H.Y. Sensor fusion method for horizon detection from an aircraft in low visibility conditions. *IEEE Trans. Instru. Meas.*, 2014, **63**(3), 620-627.  
doi: 10.1109/TIM.2013.2272843
- Zheng, Z.W.; Jin, Z.H.; Sun, L.; Zhu, M. Adaptive sliding mode relative motion control for autonomous carrier landing of fixed-wing unmanned aerial vehicle. *IEEE Access*, 2017, **5**, 5556-5565.  
doi: 10.1109/ACCESS.2017.2671440.
- Jason, T.I.; Kenan, O.E. & Joao, P.H. Local carrier-based precision approach and landing system. *In Proceeding of 55<sup>th</sup> CDC, Las Vegas, USA, 2016*, pp. 6284-6290.  
doi: 10.1109/CDC.2016.7799236
- Oleg, Vygolov & Sergey, Zheltov. Enhanced, synthetic and combined vision technologies for civil aviation. *In Computer Vision in Control System-2*, edited by Margarita N. Favorskaya & Lakhmi C. Jain. Springer International Publishing AG, Gewerbestrasse, Switzerland, 2015. pp. 201-230.  
doi: 10.1007/978-3-319-11430-9\_8
- Laiacker, M.; Kondak, K.; Schwarzbach, M. & Muskardin,

- T. Vision aided automatic landing system for fixed wing UAV. *In ICIRS*, Tokyo, Japan, 2013.  
doi: 10.1109/IROS.2013.6696777
8. Guo, P.Y.; Li, X.; Gui, Y.; Zhou, X.; Zhang, H.L. & Zhang, X.H. Airborne vision-aided landing navigation system for fixed-wing UAV. *In 12<sup>th</sup> IEEE ICSP*, Beijing, China, 2014.  
doi: 10.1109/icosp.2014.7015193
  9. Chiu, H-P.; Das, A.; Miller, P.; Samarasekera, S. & Kumar, K. Precise vision-aided aerial navigation. *In IEEE/RSJ ICIRS*, Chicago, IL, USA, 2014. pp. 688-695.  
doi: 10.1109/IROS.2014.6942633
  10. Abu-Jbara, K.; Alheadary, W.; Sundaramorthi, G. & Claudel, C. A robust vision-based detection and tracking algorithm for automatic UAV landing. *In ICUAS*, Denver, Colorado, USA, 2015. pp.1148-1157.  
doi: 10.1109/ICUAS.2015.7152407.
  11. Satish Kumar, V.; Kashyao, S.K. & Shantha Kumar, N. Detection of runway and obstacles using electro-optical and infrared sensors before landing. *Def. Sci. J.*, 2014, **64**(1), 67-76.  
doi: 10.14429/dsj.64.2765
  12. Chen, Y.Y. & Shao, Y.S. Radon transform-based algorithm for the detection and matching of airport objects. *J. Tongji Univer.*, 2006, **34**(6), 732-737. (Chinese).  
doi: 10.1007/11612704\_73
  13. Kou, Z.Y.; Shi, Z.W. & Liu, L. Airport detection based on line segment detector. *In Proceedings of ICCV in Remote Sensing*, Xiamen, China, 2012. pp. 72-77.  
doi: 10.1109/CVRS.2012.6421236.
  14. Wu, W.; Xia, R.B.; Xiang, W.; Hui, B.; Chang, Z.; Liu, Y.P. & Zhang, Y.H. Recognition of airport runways in FLIR images based on knowledge. *IEEE Geosci. Remote Sens. Lett.*, 2014, **11**(9), 1534-1538.  
doi: 10.1109/LGRS.2014.2299898
  15. Wu, W.; Xia, R.B.; Xiang, W.; Hui, B.; Chang, Z.; Liu, Y.P. & Zhang, Y.H. Efficient airport detection using line segment detector and fisher vector representation. *IEEE Geosci. Remote Sens. Lett.*, 2016, **13**(8), 1079-1083.  
doi: 10.1109/LGRS.2016.2565706
  16. Dong, Y.W.; Yuan, B.C.; Wang, H.Y. & Shi, Z.M. A runway recognition algorithm based on heuristic line extraction. *In Proceedings of 3<sup>rd</sup> ICIASP*, Wuhan, China, 2011, pp. 292-296. doi:10.1109/IASP.2011.6109049
  17. Weng, J.Y. & Hwang, W.S. Incremental hierarchical discriminant regression. *IEEE Trans. Neural Networ.*, 2007, **18**(2), 397-415.  
doi: 10.1109/TNN.2006.889942
  18. Wang, X.; Wang, B. & Zhang, L.M. Airport detection in remote sensing images based on visual attention. *In Proceedings of 18th ICNIP*, Shanghai, China, 2011, pp. 475-484.  
doi: 10.1007/978-3-642-24965-5\_54
  19. Tao, C.; Tan, Y.H.; Cai, H.J. & Tian, J.W. Airport detection from large IKONOS images using clustered SIFT keypoints and region information. *IEEE Geosci. Remote Sens. Lett.*, 2011, **8**(1), 128-132.  
doi: 10.1109/LGRS.2010.2051792
  20. Yao, X.W.; Han, J.W.; Guo, L.; Bu, Shuhui & Liu, Z.B. A coarse-to-fine model for airport detection remote sensing images using target-oriented visual saliency and CRF. *Neurocomputing*, 2015, **164**(2015) 162-172.  
doi: 10.1016/j.neucom.2015.02.073
  21. Liu, D.H.; He, L.H. & Carin, L. Airport detection in large aerial optical imagery. *In Proceedings of IEEE ICASSP*, Montreal, Quebec, Canada, 2004, pp.761-764.  
doi: 10.1109/ICASSP.2004.1327222
  22. Zhu, D.; Wang, B. & Zhang, L.M. Airport target detection in remote sensing images: a new method based on two-way saliency. *IEEE Geosci. Remote Sens. Lett.*, 2015, **12**(5), 1096-1100.  
doi: 10.1109/LGRS.2014.2384051
  23. Tang, G.F.; Xiao, Z.F.; Liu, Q. & Liu, H. A novel airport detection method via line segment classification and texture classification. *IEEE Geosci. Remote Sens. Lett.*, 2015, **12**(12), 2048-2412.  
doi: 10.1109/LGRS.2015.2479681
  24. Aytekin, O.; Zongur, U. & Halici, U. Texture-based airport runway detection. *IEEE Geosci. Remote Sens. Lett.*, 2013, **10**(3), 471-475.  
doi: 10.1109/LGRS.2012.2210189
  25. Wang, W.; Liu, L.; Hu, C.B.; Jiang, Y.M. & Kuang, G.Y. Airport detection in SAR image based on perceptual organization. *In Proceedings of International Workshop on M2RSM*, Xiamen, China, 2011, pp. 1-5.  
doi: 10.1109/M2RSM.2011.5697415
  26. Bhagavathy, S. & Manjunath, B.S. Modeling and detection of geospatial objects using texture motifs. *IEEE Trans. Geosci. Remote Sens.*, 2006, **44**(12), 3706-3715.  
doi: 10.1109/TGRS.2006.881741
  27. Zhuang, H.L. & Low, K.S. Real time runway detection in satellite images using multi-channel PCNN. *In Proceedings of 9<sup>th</sup> ICIEA*, Hangzhou, China, 2014, pp. 253-257.  
doi: 10.1109/ICIEA.2014.6931168
  28. Gong, X.J.; Abbott, A.L. & Fleming, G.A. A survey of techniques for detection and tracking of airport runways. *In 44<sup>th</sup> AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, USA, 2006. pp.17289-17302.
  29. Zhang, J.W. & Peng, Z.M. An improved template tracking method based on rigid extended object. *In Proceedings of 5<sup>th</sup> ISPD*, Beijing, China, 2013. pp. 890717.1-10.  
doi: 10.1117/12.2032254.
  30. Jain, A.K.; Zhong, Y. & Lakshmanan, S. Object matching using deformable templates. *IEEE Trans. Pattern Anal.*, 1996, **18**(3), 267-278.
  31. Zuo, Z.R.; Chen, H.F. & Zhang, T.X. Study on hit-aim detection of airfield runway based on weighted structure templates matching. *In Proceedings of SPIE on MIPPR*, 2009. pp.749532.1-8.  
doi: 10.1117/12.834006
  32. Hartley, R. & Zisserman, A. Multiple view geometry in computer vision. Cambridge Univ. Press, Cambridge, UK, 2003. pp.153-158.
  33. Grompone Von Gioi, R.; Jakubowicz, J.; Morel, J.M. & Randall, G. On straight line segment detection. *J. Math.*

- Imaging Vis.*, 2008, **32**(3), 313-347.  
doi: 10.1007/s10851-008-0102-5
34. Grompone Von Gioi, R.; Jakubowicz, J.; Morel, J. M. & Randall, G. LSD: A fast line segment detector with a false detection control. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2010, **32**(4), 722–732.  
doi: 10.1109/TPAMI.2008.300
  35. Grompone Von Gioi, R.; Jakubowicz, J.; Morel, J.M. & Randall, G. LSD: a line segment detector. *Image Processing On Line*, 2012-03-24, 35-55.  
doi:10.5201/ipol.2012.gjmr-lsd
  36. Akinlar, C. & Topal, C. EDLines: A real-time line segment detector with a false detection control. *Pattern Recognition Lett.*, 2011, **32**, 1633-1642.  
doi: 10.1016/j.patrec.2011.06.001
  37. Akinlar, C. & Topal, C. EDLines: real-time line segment detection by edge drawing (ED). In 18<sup>th</sup> IEEE ICIP, Brussels, Belgium, 2011, pp. 2837-2840.  
doi: 10.1109/ICIP.2011.6116138
  38. Akinlar, C. & Topal, C. EDPF: a real-time parameter-free edge segment detector with a false detection control. *Int. J. Pattern Recogn.*, 2012, **26**(1), 1-22.  
doi: 10.1142/S0218001412550026
  39. Topal, C.; Akinlar, C. & Genc, Y. Edge drawing: A heuristic approach to robust real-time edge detection. In Proceedings of 20<sup>th</sup> ICPR, Istanbul, Turkey, 2010. pp. 2424-2427.  
doi: 10.1109/ICPR.2010.593
  40. Topal, C. & Akinlar, C. Edge drawing: a combined real-time edge and segment detector. *J. Vis. Commun. Image R.*, 2012, **23**, 862-872.  
doi: 10.1016/j.jvcir.2012.05.004
  41. Liu, D.; Wang, Y.T.; Tang, Z. & Lu, XQ. A robust and fast line segment detector based on top-down smaller eigenvalue analysis. In Fifth ICGIP, Hongkong, China, 2013, pp.1-5.  
doi: 10.1117/12.2050864
  42. Dai, F. & Zhu, Z.H. Line segment grouping and linking: a key step toward automated photogrammetry for non-contact site. *J. Intell. Robot Syst.*, 2015, **79**, 371-384.  
doi: 10.1007/s10846-014-0119-5

## ACKNOWLEDGEMENTS

This work was supported by the AVIC Technology Innovation Foundation under Grant No.2014D63130R and the Aviation Science Foundation under Grant No.2014ZC31004.

## CONTRIBUTORS

**Mr Lei Zhang** received the BS (Computer Science) from Xi'an Jiao Tong University, in 2001, and MS from China Aviation Academe, in 2011. Now he is a senior engineer in Xi'an Aeronautic Computing Technique Research Institute, and also pursuing his PhD in Computer Science and Technology at Northwestern Polytechnical University. His research interests include : Computer vision, image processing and vision based navigation in airborne system.

In this current study, he propose a novel runway detection method based on airborne multi-sensors data fusion which works in a coarse-to-fine hierarchical architecture, then design and implement this algorithm, verify the method by experiments, and finally write this paper.

**Dr Yue Cheng** received the BS and PhD in Electronic Engineering from Zhejiang University, in 2008 and 2013, respectively. And post doctor from Northwestern Polytechnical University, in 2014. Currently working as a senior engineer in Xi'an Aeronautic Computing Technique Research Institute. His research interests include : Computer vision and augmented reality in airborne system.

In this current study, he take advantage of virtual reality simulation development software (VT MAK Simulator) to generate visible and infrared video corresponding with airborne navigation parameters to support experiments.

**Prof. Zhengjun Zhai** received his BS (Information System Engineering) and MS (Computer Science and Technology) from Northwestern Polytechnical University, in 1985 and 1988, respectively. Now he is the Director of Computer Measurement and Control and Simulation Technology Institute in Northwestern Polytechnical University. His research interests include : Information processing, embedded system and avionics system.

In this current study, he review the algorithm design, experiments and results. In addition, He reviews the draft of the paper and give some suggestion for improvement.