Detection and Tracking Strategies for Autonomous Aerial Refuelling Tasks Based on Monocular Vision

Regular Paper

Yingjie Yin[1], De Xu[1,*], Xingang Wang[1] and Mingran Bai[1]

1 Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, China
* Corresponding author E-mail: de.xu@ia.ac.cn

Abstract Detection and tracking strategies based on monocular vision are proposed for autonomous aerial refuelling tasks. The drogue attached to the fuel tanker aircraft has two important features. The grey values of the drogue's inner part are different from the external umbrella ribs, as shown in the image. The shape of the drogue's inner dark part is nearly circular. According to crucial prior knowledge, the rough and fine positioning algorithms are designed to detect the drogue. Particle filter based on the drogue's shape is proposed to track the drogue. A strategy to switch between detection and tracking is proposed to improve the robustness of the algorithms. The inner dark part of the drogue is segmented precisely in the detecting and tracking process and the segmented circular part can be used to measure its spatial position. The experimental results show that the proposed method has good performance in real-time and satisfied robustness and positioning accuracy.

Keywords Positioning, Monocular Vision, Visual Detection, Visual Tracking, Particle Filter, Aerial Refuelling

1. Introduction

Aerial refuelling, also referred to as in-flight refuelling (IFR) or air-to-air refuelling (AAR), is an operation whereby fuel is transferred from one aircraft (the tanker) to another aircraft (the receiver) during flight. IFR is an important method for extending the flying distance and speed of the aircraft and is widely used in military aircraft. In unmanned aerial vehicles (UAV), autonomous air-to-air refuelling is needed to ensure flight endurance. There are two kinds of hardware configurations used for aerial refuelling: the first configuration, called the boom-and-receptacle refuelling system, includes a rigid boom extending from the tanker aircraft, with a probe and nozzle at its distal end. The boom also includes airfoils controlled by a boom operator stationed on the refuelling aircraft. The airfoils allow the boom operator to actively manoeuvre the boom with respect to the receiver aircraft, which flies in a fixed refuelling position below and aft of the tanker aircraft [1-5]. The second configuration, called the probe-and-drogue refuelling system, includes a refuelling hose which has a drogue deposited at its end trailed behind the tanker aircraft and a probe installed on

the receiver aircraft. The probe must be placed or docked into the drogue in order to refuel successfully [6-9]. Autonomous aerial refuelling relies on three key technologies: target detection, tracking and measurement, in order to allow the receiver aircraft to determine control strategies to enable a robust and safe approach and coupling. The attempt described in this paper is to provide detection and tracking strategies for the probe-drogue autonomous aerial refuelling based on monocular vision.

In this paper, the drogue's detection and tracking strategies based on monocular vision are proposed for autonomous aerial refuelling tasks. Two important features of the drogue are used to design the detection and tracking strategies. The first feature is that the grey values of the drogue's inner part are almost the same and are different from the external umbrella ribs. The second feature is that the shape of the drogue's inner dark part is nearly circular. The drogue's detection algorithm includes two parts: the drogue's rough location algorithm and the drogue's fine positioning algorithm. The rough location algorithm is used to define the potential regions in which the drogue may be located, while the drogue's fine positioning algorithm is used to find the drogue in the potential regions accurately if the drogue is in the image. Particle filter is widely used in target tracking because of its robustness [10-13]. A new particle filter algorithm based on the drogue's shape was proposed to track the drogue. In the new particle algorithm, unique principles of state transition are defined to ensure tracking robustness even when the drogue's position or size changes significantly in two adjacent frames. A switch strategy between detection and tracking was proposed to improve the algorithm's robustness, which provides the link between detection and tracking. This is critical when the tracking has failed or the drogue is not in the image.

The paper is organized as follows. Section 2 gives an introduction to previous works. Section 3 describes the drogue's detection strategy. Section 4 describes the drogue's tracking strategy. Section 5 describes the switch strategy between detection and tracking. Section 6 presents the experimental results and Section 7 concludes the main points of the research.

2. Previous Works

Machine vision methods used for autonomous aerial refuelling tasks are becoming increasingly popular [6,7,14-17]. Advantages of using machine vision methods for autonomous aerial refuelling tasks include the potential for installation without modification being required to the target aircraft and increased measurement precision.
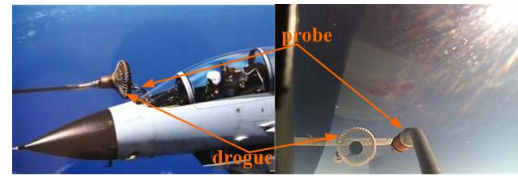

Figure 1. The probe-and-drogue refuelling system

The researchers have developed a variety of different machine vision method for the probe-drogue autonomous aerial refuelling system (shown in Figure 1). John Valasek et al. [6] developed a vision-based navigation sensor and system for autonomous aerial refuelling tasks. For application to the endgame docking problem of automated aerial refuelling of aircraft, a VisNav sensor (a position-sensing diode) was mounted on a receiver aircraft and a set of LED beacons were mounted on a drogue being trailed from a tanker aircraft. When light energy from an individual beacon on the drogue was focused on the surface of the position-sensing diode, it generated an electrical current, which was measured with four pickoff leads, one on each side. The six-degrees-of-freedom position and attitude of the sensor aircraft with respect to the drogue can be computed by the four position-sensing signals. The main disadvantage of the method proposed in [6] is that some modifications to the tanker equipment must be made to provide electrical power for beacons, since there is no such power in the hose to which the drogue is attached. Fravolini et al. [18] proposed a docking control scheme for autonomous aerial refuelling of UAVs using a probe-drogue refuelling system. The docking control scheme was based on a fuzzy sensor fusion strategy featuring GPS and machine vision data. The GPS was used to measure the relative position between the tanker and the receiver and the machine vision was used to measure the relative camera-drogue distance. Some markers were placed in the drogue to measure its position and orientation. However, GPS receivers may be affected by interference from electronic devices and GPS signals may be blocked by the tanker. Lorenzo Pollini et al. [19] placed light emitting diodes (LEDs) on the drogue and used a CCD webcam with an infra-red filter to identify the LEDs. Hager and Mjolsness's (LHM) algorithm [20] was used to determine iteratively the translation vector as well as the transformation matrix between the 3D reference systems on the object and the camera, respectively. As in [6], the main disadvantage in [19] is that some modifications to the tanker equipment are required. Carol Martinez et al. [7] proposed a vision-based strategy for autonomous aerial refuelling tasks. The proposed strategy consisted of four stages: detection, initialization, tracking and 3D position estimation. The detection stage was composed of two algorithms: one based on edge-image template matching using the normalized cross correlation (NCC) method, and the second based on image threshold segmentation. The detection method is time-consuming

because the drogue images with different variations, such as scale, illumination and position, must be contained in the template images. It is impossible to contain all the conditions of the drogue in the template images, so in order to decrease the failure of detection, an experience threshold was used to segment the image to detect the inner part of the drogue when the edge-image template matching method failed. However, the experience threshold is hard to define because the illumination of the scene may change significantly. The tracking algorithm was a Hierarchical Multi-Parametric and Multi-Resolution implementation of the Inverse Compositional Image Alignment technique HMPMR-ICIA [21].

3. Detection Strategy

3.1 Rough Location of the Drogue

The aim of rough location is to define the potential regions in which the drogue may be located. According to prior knowledge, the rough location stage is composed of two algorithms: this first based on image segmentation using a series of thresholds, and the second based on contour features of the image regions segmented by image segmentation using a series of thresholds.

It is impossible to define an accurate experience threshold used to segment the image to detect the inner part of the drogue because the illumination of the scene may change significantly. So a series of thresholds are used to segment the same image, as follows:

$$F = \bigcup_{j=0}^{N-1} g_i \delta(i - T_j) \qquad (1)$$

$$g_i(x,y) = \begin{cases} 1, & f(x,y) > i \\ 0, & f(x,y) \le i \end{cases} \qquad (2)$$

$$\delta(t) = \begin{cases} 1, & t = 0 \\ 0, & t \ne 0 \end{cases} \qquad (3)$$

where f is the input images, F is the set of output images which include $\{g_{T0}, g_{T1}, \ldots, g_{TN-1}\}$. $\{T_0, T_1, \ldots, T_{N-1}\}$ are a series of thresholds used to segment the input image, as follows:

$$T_k = T_0 + k\Delta T \qquad (4)$$

where ΔT is an increment of the threshold.

Then all the contours of the output images are extracted, and the set of the contours is expressed as $C = \{c_0, c_1, \ldots, c_{n-1}\}$. The contour of the inner part of the drogue is nearly circular, so aspect ratios of the minimum enclosing rectangles of contours are used to define the potential contours of the drogue from the set C, as follows:

$$Ratio(c_i) = \frac{\min(L(c_i), W(c_i))}{\max(L(c_i), W(c_i))} \qquad (5)$$

$$C' = \bigcup_{i=0}^{n-1} c_i u(Ratio(c_i) - T_{Ratio}) U_L U_W \qquad (6)$$

$$U_L = u(L(c_i) - T_{L1}) - u(L(c_i) - T_{L2}) \qquad (7)$$

$$U_W = u(W(c_i) - T_{W1}) - u(W(c_i) - T_{W2}) \qquad (8)$$

$$u(t) = \begin{cases} 0, & t < 0 \\ 1, & t \ge 0 \end{cases} \qquad (9)$$

where $Ratio(c_i)$ is the aspect ratios of the minimum enclosing rectangle of the contour and T_{Ratio} is threshold of the aspect ratios of the minimum enclosing rectangle. $L(c_i)$ is the length of the minimum enclosing rectangle, T_{L1} is the lower bound of the length of the minimum enclosing rectangle and T_{L2} is the upper bound of the length of the minimum enclosing rectangle. $W(c_i)$ is the width of the minimum enclosing rectangle, T_{W1} is the lower bound of the width of the minimum enclosing rectangle and T_{W2} is the upper bound of the width of the minimum enclosing rectangle. $C' = \{c'_0, c'_1, \cdots, c'_{n'-1}\}$ is the set of the drogue's potential contours, that is to say, the position in which the drogue is located can be obtained from the positions of C'.

In order to improve the speed of detection, the Multi-Resolution (MR) hierarchical structure [22] is used. The MR structure is created by repeatedly downsampling the images by a factor of two in order to create the different levels of the pyramid. The number of levels pL is defined, taking into account the size of the drogue in the image. The general idea of the acceleration strategy is that the rough location of the drogue is conducted at the lowest resolution level. The advantage of using the MR structure is that many small error contours will not be segmented out at low resolutions.

3.2 Fine Positioning of the Drogue

3.2.1 Location of Edge Points of the Drogue's Inner Dark Part

To every contour in the set of C', the circular least-square fitting [23] is used to obtain the contours' centres, as follows:

$$\begin{aligned} c^i_{cent\,X} &= \frac{A-B}{E} \\ c^i_{cent\,Y} &= \frac{C-D}{E} \\ c^i_{cent\,R} &= \sqrt{c^{i\,2}_{cent\,X} - 2\overline{x}c^i_{cent\,X} + c^{i\,2}_{cent\,Y} - 2\overline{y}c^i_{cent\,Y} + \overline{x^2} + \overline{y^2}} \end{aligned} \qquad (10)$$

$$\begin{aligned} A &= (\overline{x^2 x} + \overline{x y^2} - \overline{x^3} - \overline{xy^2})(\overline{y^2} - \overline{y}^2) \\ B &= (\overline{x^2 y} + \overline{y y^2} - \overline{x^2 y} - \overline{y^3})(\overline{xy} - \overline{x}\,\overline{y}) \\ C &= (\overline{x^2 y} + \overline{y y^2} - \overline{x^2 y} - \overline{y^3})(\overline{x}^2 - \overline{x^2}) \\ D &= (\overline{x^2 x} + \overline{x y^2} - \overline{x^3} - \overline{xy^2})(\overline{xy} - \overline{x}\,\overline{y}) \\ E &= 2(\overline{x}^2 - \overline{x^2})(\overline{y}^2 - \overline{y^2}) - 2(\overline{xy} - \overline{x}\,\overline{y})^2 \\ \overline{x^m y^n} &= \sum_{i \in c'_i} x_i^m y_i^n \Big/ \sum_{i \in c'_i} 1 \end{aligned} \qquad (11)$$

where (c^i_{centX}, c^i_{centY}) are the coordinates of the centre of the contour c'_i and (x_i, y_i) is the point in the contour c'_i. Some contours in the set $C' = \{c'_0, c'_1, \cdots, c'_{n'-1}\}$ may belong to the same object because a series of thresholds are used to segment the same image. Thus, the distance between different contours in the set C' can be used to eliminate unnecessary contours, as follows:

$$d_{ij} = \sqrt{\left(c^i_{cent\,X} - c^j_{cent\,X}\right)^2 + \left(c^i_{cent\,Y} - c^j_{cent\,Y}\right)^2}$$
$$C^l = \bigcup_{\substack{i \neq j \\ i,j=0}}^{n'-1} c'_i u\left(d_{ij} - T_d\right) \qquad (12)$$

where T_d is a threshold for eliminating unnecessary contours, $C^l = \{c^l_0, c^l_1, \cdots, c^l_{n'-1}\}$ is the set of contours after eliminating unnecessary contours and the function $u(t)$ is defined in (9). The set of the centres of contours C^l can be expressed as $C^c = \{c^0_{cent}, c^1_{cent}, \cdots, c^{n'-1}_{cent}\}$.
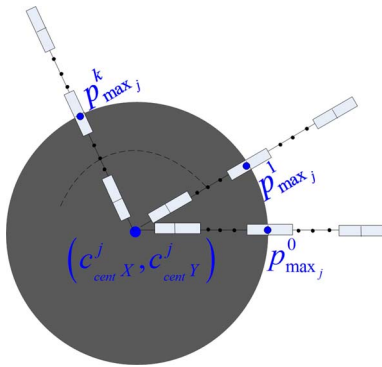

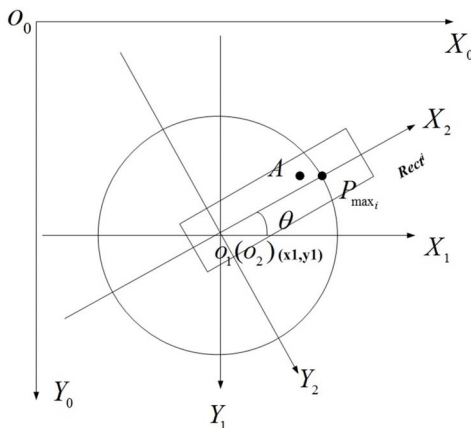**Figure 2.** Location of edge points of the drogue's inner dark part.

Figure 3. Three coordinate systems

In order to obtain the location of edge points of the drogue's inner dark part, some half-lines are assumed to start at the point ($2^{(pL-1)} * c^i_{centX}$, $2^{(pL-1)} * c^i_{centY}$) in the set C^c at the highest resolution level, and extend outward. The amplification coefficient from the lowest resolution level to the highest resolution level is $2^{(pL-1)}$, and pL is the

number of levels of the pyramid. The angles between adjacent half-lines are equal. A small rectangle moves along every half-line, and the small rectangle is divided into two equal rectangles. The one near the point ($2^{(pL-1)} * c^i_{centX}$, $2^{(pL-1)} * c^i_{centY}$) is called the interior rectangle, while the other, further from the point ($2^{(pL-1)} * c^i_{centX}$, $2^{(pL-1)} * c^i_{centY}$), is called the exterior rectangle. The ratios of the sum of the pixel values in the exterior rectangle to the sum of the pixel values in the interior rectangle are calculated when the small rectangle is moved along every half-line. All the ratios in the same half-line are compared to each other to find the position in which the ratio of the small rectangle's two parts is maximal, as follows:

$$p_{\max_k} = \arg\max_{p \in half-line_k} \{Ratio(p)\} \qquad (13)$$

where p is the point in the half-line k, $Ratio(p)$ is the ratio of the point p in the half-line k and $p_{\max_k}$ is the position corresponding to the maximal ratio in the half-line k. Then, the edge points corresponding to the jth contour in the set C^l or the jth centre in the set C^c can be expressed as $P_{\max_j} = \left\{p^0_{\max_j}, p^1_{\max_j}, \cdots, p^{n^k_p-1}_{\max_j}\right\}$.

Most of directions of the half-lines are neither horizontal nor vertical, so coordinate transformations are used to calculate the coordinates of small rectangles in half-lines whose directions are not horizontal or vertical. Figure 3 shows three coordinate systems; the coordinate system (X_0, Y_0) is centred on the upper-left corner of the image, the X_0 axis points to the right horizontally and the Y_0 axis points vertically downwards. The coordinate system (X_1, Y_1) is centred at the point (x_1, y_1) which is the translation relative to the point O_0. The X_1 axis is parallel to the X_0 axis and the Y_1 axis is parallel to the Y_0 axis, but their directions are opposite. The coordinate system (X_2, Y_2) is centred at the point (x_1, y_1). The X_2 axis overlaps with the half-line i and the Y_2 axis is perpendicular to the X_2 axis.

The coordinates of the point A are (a_0, b_0) in the coordinate system (X_0, Y_0), (a_1, b_1) in the coordinate system (X_1, Y_1) and (a_2, b_2) in the coordinate system (X_2, Y_2). The relationship between (a_1, b_1) and (a_2, b_2) is:

$$\begin{bmatrix} a_1 \\ b_1 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \\ 1 \end{bmatrix} \qquad (14)$$

and the relationship between (a_0, b_0) and (a_1, b_1) is:

$$\begin{bmatrix} a_0 \\ b_0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_1 \\ 0 & 1 & y_1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \\ 1 \end{bmatrix} \qquad (15)$$

From (14) and (15), the relationship between (a_0, b_0) and (a_2, b_2) can be obtained as follows:

$$\begin{bmatrix} a_0 \\ b_0 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & x_1 \\ -\sin\theta & \cos\theta & y_1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \\ 1 \end{bmatrix} \tag{16}$$

The parameter θ is the angle between the X_1 axis and the X_2 axis. The coordinates of points in the coordinate system (X_0, Y_0) can be calculated by equation (16) if the coordinates in the coordinate system (X_2, Y_2) are known in advance.
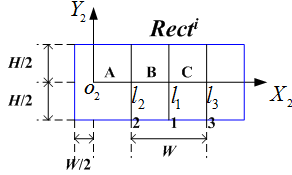

Figure 4. The rectangle $Rect^i$ used to calculate the integral image

In order to improve the computing speed, the integral image is used to calculate the ratio of the small rectangle's two parts. The $Rect^i$ in Figure 3 and Figure 4 is the rectangle that is used to calculate the integral image in the half-line i. The height of the rectangle $Rect^i$ is the same as the heights of the small rectangles which move along the half-lines in Figure 2: all are H. The widths of the small rectangles are W and the width of the rectangle $Rect^i$ is $W+N_s$. N_s is the moving time of the small rectangle in every half-line. The integral image of the rectangle $Rect^i$ is calculated as follows:

$$\begin{aligned} s(y) &= \sum_{x=0}^{H} i(x, y) \\ ii(H, y) &= ii(H, y-1) + s(y) \end{aligned} \tag{17}$$

where $s(y)$ is the sum of the y-th column pixels, $ii(H, y)$ is the value of the last line of the integral image. In Figure 4, l_1 is the boundary of the small rectangle's two parts, l_2 is the left side of the small rectangle and l_3 is the right side of the small rectangle. The value of the integral image at location 2 is the sum of the pixels in rectangle A. The value of the integral image at location 1 are the sum of the pixels in rectangle $A+B$ and the value of the integral image at location 3 is the sum of the pixels in rectangle $A+B+C$. Therefore, the sum of the pixels in rectangle B is $ii_1 - ii_2$ and the sum of the pixels in rectangle C is $ii_3 - ii_1$. The ratio of the small rectangle's two parts is ($ii_3 - ii_1$)/ ($ii_1 - ii_2$), while ii_1, ii_2, ii_3 are the values of the integral image at locations 1, 2 and 3 respectively.

3.2.2 Getting Rid of the Bad Edge Points Using Vector Angles

The bad edge points of the drogue's inner dark part may be detected because of image noise or the partial occlusion of the drogue. Assume that the detected edge points of the drogue's inner dark part are the red points {1, 2, …, 16} shown in Figure 5. We define the vectors as {$\overrightarrow{1\ 2}$, $\overrightarrow{2\ 3}$, …, $\overrightarrow{15\ 16}$, $\overrightarrow{16\ 1}$}. As shown in Figure 5, the

vector angle θ_1 of the detected edge point 1 is the angle between the vector $\overrightarrow{16\ 1}$ and the vector $\overrightarrow{1\ 2}$. The vector angle θ_7 of the detected edge point 7 is the angle between the vector $\overrightarrow{6\ 7}$ and the vector $\overrightarrow{7\ 8}$. In the same way, we can obtain other vector angles. The types of the bad edge points are diverse, but the bad edge points which own larger vector angles are representative. In Figure 5, the points 4, 8, 13 and 15 are the typical bad edge points which produce larger vector angles. The vector angles can therefore be used to get rid of the bad edge points, as follows:

$$P = \bigcup_{i=0}^{n_p-1} p_{\max_i} u(|\theta_T| - |\theta_i|) \tag{18}$$

where $P=\{p_0, p_1, …, p_{n1}\}$ is the set of the remaining edge points after getting rid of the bad edge points using vector angles, n_p is the number of the edge points detected in Section 3.2.1, n_1 is the number of the remaining edge points after getting rid of the bad edge points using vector angles, θ_T is the threshold of the vector angles, θ_i is the vector angle of the edge point and the function $u(t)$ is defined in (9). The bad edge points 4, 8, 13 and 15 in Figure 5 can be eliminated by equation (18).
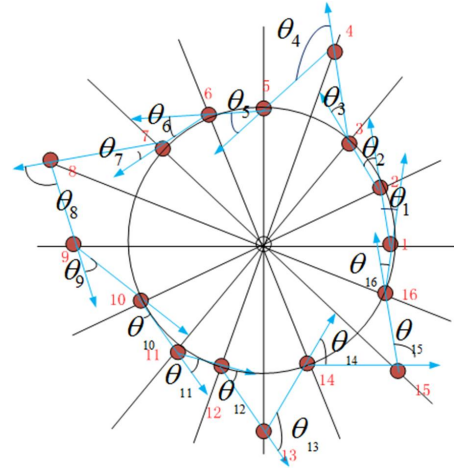

Figure 5. Vector angles of edge points

3.2.3 Location of the Drogue Using RANSAC

The styles of the bad edge points are diverse, so some bad edge points cannot be eliminated by vector angles. For example, the bad point 9 in Figure 5 cannot be eliminated by the vector angle because the vector angle θ_9 is less than the threshold θ_T of the vector angles. The shape of the drogue's inner dark part is nearly circular, so good edge points can be determined by this prior knowledge. RANSAC [24], an abbreviation for "random sample consensus", is an iterative method to estimate parameters of a mathematical model from a set of observed data which contains outliers. Each contour c_i^l in the set of C^l corresponds to a set $P^i=\{p^i_0, p^i_1, …, p^i_{n1}\}$ of the remaining edge points after getting rid of the bad edge points using

```
Input: The number of iterations K, the number of samples used to fit the model
       n_s, the threshold T_r, the edge points P^i = {p_0^i, p_1^i, ···, p_{n1}^i} corresponding to
       the contour c_i^l and the threshold T_W
Output: The state of detection S, the center of the inner dark part of the drogue
        (u,v) and the radius of the inner dark part of the drogue r

1  for each P^i corresponding to the contour c_i^l do
2      for k = 1 to K do
3          n_s points are randomly selected from P^i, W_k^i=0;
4          Calculated the circle center (u_k,v_k) and the radius r_k according to the
           circular least-square fitting;
5          for each p_j^i point in the set P^i do
6              Calculate the distance d between the point p_j^i and the circle center
               (u_k,v_k), Δr = |d − r_k|;
7              if Δr < T_r then
8                  W_k^i = W_k^i + 1;
9              end
10         end
11     end
12     k* = arg max_{k∈[1,K]}{W_k^i}, W_max^i = W_k*^i, u_i = u_k*, v_i := v_k*, r_i = r_k*;
13 end
14 n* = arg max_{i∈[0,n'−1]}{W_max^i}, W_max = max_{i∈[0,n'−1]}{W_max^i};
15 if W_max ≥ T_W then
16     S=1, u = u_n*, v = v_n*, r = r_n*;
17     Return S, (u,v) and r;
18 end
19 else
20     S=0;
21     Return S;
22 end
```

Figure 6. Location of the drogue using RANSAC

vector angles. The good edge points of P^i of the contour c_i^l can be determined by the RANSAC method. The weight W^i_{max} is defined as the number of good edge points of P^i of the contour c_i^l and W^i_{max} can be explained as a posteriori probability, also expressible as the probability of the contour c_i^l being the contour of the drogue. Then the contour c_*^l with the highest probability W_{max} is identified, as follows:

$$W_{max} = \max_{i \in [0, n'-1]}\{W_{max}^i\}$$
$$n^* = \arg \max_{i \in [0, n'-1]}\{W_{max}^i\} \qquad (19)$$
$$c_*^l = c_{n^*}^l$$

A threshold T_W is used to define whether the set C^l includes the contour of the drogue, as follows:

$$S = \begin{cases} 1 & W_{max} \geq T_W \\ 0 & W_{max} < T_W \end{cases} \qquad (20)$$

where S is the state of detection, $S=1$ indicates there is a drogue in the image and $S=0$ indicates there isn't a drogue in the image. The pseudo-code of the algorithm for location of the drogue using RANSAC is presented in Figure 6.

4. Tracking Strategy

The tracking algorithm is based on particle filters which are sequential Monte Carlo methods [25] based on point

mass. Particle filters are suitable for any non-linear system that could be represented by a state model. The tracking object is the inner dark part of the drogue in the tracking algorithm, so the centre of the inner dark part of the drogue can be defined as the state x_t of the drogue at time t.

4.1 Selection of Particles and State Transition

The disturbance of the drogue is uncertain when in the air, so it is hard to establish the accurate motion model of the drogue during air-to-air refuelling. The state x_{t-1} of the drogue at time t-1 is selected as the particle at time t. The number of the particle is N_{in}+ N_{ex}. N_{in} is the number of particles called interior particles whose range of state transition is limited to the interior of the circle C_{cir} whose centre is x_{t-1} and radius is r, while N_{ex} is the number of particles called exterior particles whose range of state transition is limited to some half-lines in the exterior of the circle C_{cir}. In Figure 7, the half-lines to which exterior particles whose range of state transition is limited are assumed to start at the position which is l away from the centre x_{t-1}, and extend outward. The angle between two adjacent half-lines is θ and the distance between two adjacent exterior state transition particles in the same half-line is Δs. The process of state transition of inner particles is as follows:

$$x_t^{k_{in}} = \left(X_t^{k_{in}}, Y_t^{k_{in}} \right)$$
$$X_t^{k_{in}} = \begin{cases} X_{t-1} & k_{in} = 0 \\ X_{t-1} + R_{ran}\cos\theta_{ran} & 0 < k_{in} < N_{in} \end{cases} \qquad (21)$$
$$Y_t^{k_{in}} = \begin{cases} Y_{t-1} & k_{in} = 0 \\ Y_{t-1} + R_{ran}\sin\theta_{ran} & 0 < k_{in} < N_{in} \end{cases}$$

where $x_t^{k_{in}}$ is the interior state transition particle, $\left(X_t^{k_{in}}, Y_t^{k_{in}} \right)$ is the coordinate of $x_t^{k_{in}}$, R_{ran} is a random variable which obeys a uniform distribution $U(0, r)$, θ_{ran} is a random variable which obeys a uniform distribution $U(0, 2\pi)$ and (X_{t-1}, Y_{t-1}) are the coordinates of x_{t-1}. The process of state transition of exterior particles is as follows:
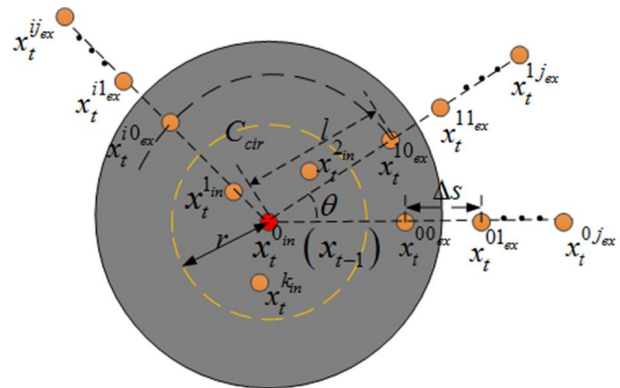
Figure 7. Principles of state transition

$$x_t^{ij_{ex}} = \left(X_t^{ij_{ex}}, Y_t^{ij_{ex}} \right)$$
$$X_t^{ij_{ex}} = (l + j * \Delta s) \cos(i * \theta) \qquad (22)$$
$$Y_t^{ij_{ex}} = (l + j * \Delta s) \sin(i * \theta)$$

where $x_t^{ij_{ex}}$ is the exterior state transition particle which is the jth particle in the ith half-line, $\left(X_t^{ij_{ex}}, Y_t^{ij_{ex}} \right)$ is the coordinate of $x_t^{ij_{ex}}$.

4.2 Particle weights and posterior probability

The particle weights of state transition particles in Section 4.1 can be calculated through the following steps:

1) Obtain the set $P_{\max_j} = \left\{ p_{\max_j}^0, p_{\max_j}^1, \cdots, p_{\max_j}^{n_p-1} \right\}$ corresponding to the jth state transition particles with the same technique in Figure 2.

2) Eliminate the bad edge points of every set $P_{\max_j}$ using vector angles with the same method in Section 3.2.2, and the remaining edge points of the set $P_{\max_j}$ can be expressed as $P_j = \left\{ p_j^0, p_j^1, \cdots, p_j^{n_1-1} \right\}$.

3) Calculate weights of state transition particles using the RANSAC method with the same method in Section 3.2.3, and the weight of jth state transition particle can be expressed as W_{max}^j.

4) A threshold T_P is used to get rid of bad state transition particles as follows:

$$W_{max}^j = \begin{cases} W_{max}^j & W_{max}^j \geq T_P \\ 0 & W_{max}^j < T_P \end{cases} \qquad (23)$$

5) Calculate the state weight at time t and normalize the weights of state transition particles as follows:

$$W^t = \max_{i=0}^{N_{in}+N_{ex}-1} (W_{max}^i)$$
$$W_{max}^j = \frac{W_{max}^j}{\sum_{i=0}^{N_{in}+N_{ex}-1} W_{max}^i} \qquad (24)$$

where W^t is the weight of the state x_t used to access the performance of tracking. The posterior probability at time t can be understood as the drogue's state x_t and the drogue's state x_t can be calculated as follows:

$$x_t = \left(X_t, Y_t \right)$$
$$X_t = \sum_{i=0}^{N_{in}+N_{ex}-1} W_{max}^i X_t^i$$
$$Y_t = \sum_{i=0}^{N_{in}+N_{ex}-1} W_{max}^i Y_t^i \qquad (25)$$
$$r_t = \sum_{i=0}^{N_{in}+N_{ex}-1} W_{max}^i r_t^i$$

where (X_t, Y_t) are the coordinates of the drogue's state x_t, (X_t^i, Y_t^i) are the coordinates of the state transition particles, r_t^i is the radius calculated with the same method

in Section 3.2.3 corresponding to the ith state transition particles and r_t is the radius of the inner dark part of the drogue at time t.

5. Switch Strategy between Detection and Tracking

The detection stage must be enabled automatically to detect the drogue either at the start of the run or when the drogue has gone out of the field of view of the camera, or alternatively because the tracking algorithm has failed to track the drogue. Therefore, performance assessment criteria should be defined to switch between detection algorithm and tracking algorithm. The detection and tracking algorithm is initiated with a lost status $L=1$(i.e.. no drogue has been detected). The detection algorithm is then enabled to find the drogue. The drogue is detected successfully when the state of detection $S=1$ in Section 3.2.3, then the lost status is $L=0$ and the tracking algorithm is enabled. The performance assessment criteria of the tracking algorithm can be defined according to the weights of the drogue's states in k_s successive frames as follows:

$$L = \delta\left(\sum_{i=t}^{t+k_s-1} u(W^i - T_t) \right) \qquad (26)$$

where $\delta(t)$ is defined in (3), $u(t)$ is defined in (9), W^i is the state weight at time I and T_t is a fixed threshold. If the lost status is $L=0$, the tracking algorithm continues running. If the lost status is $L=1$, the tracking algorithm is stopped and the detection stage is enabled in the region of interest (ROI) of the image. The lost status is $L=0$ and the tracking
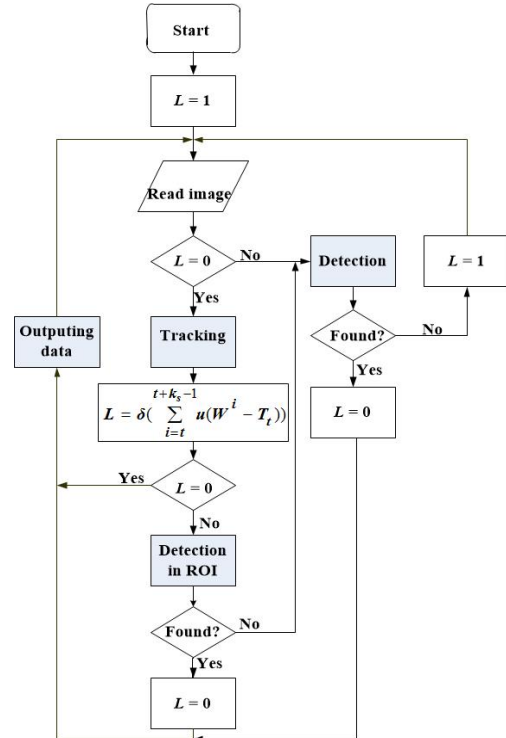

Figure 8. The proposed visual detecting and tracking system for air-to-air refuelling

algorithm is enabled if the drogue is detected successfully, otherwise the lost status is $L=1$ and the detection stage is enabled in all regions of the image. The process of the strategy for switching between detection and tracking is shown in the proposed visual detecting and tracking system for air-to-air refuelling in Figure 8.

6. Experimental Results

In this section, experiments were conducted on the real drogue of air-to-air refuelling at different air scenes. The performance of our method was compared to the performance of the algorithm proposed by Carol Martinez et al. [7]. Three experiments were carried out to detect and track different drogues at different air scenes. Speed of processing and percentage of correct location are compared between Carol Martinez's method and ours. The speed indicators were the average time t_{fave} of processing each image, the maximum time t_{max}, the minimum time t_{min} and the average time t_{ave} between the adjacent outputs when the drogue was in the image. The percentages of correct location are compared between Carol Martinez's method and ours at different location error thresholds. The proposed algorithm was developed in C++ and the OpenCV libraries were used for managing image data and the experiments were carried out on a PC with a AMD Athlon (tm) II X4 645 Processor and a 3.1GH clock.
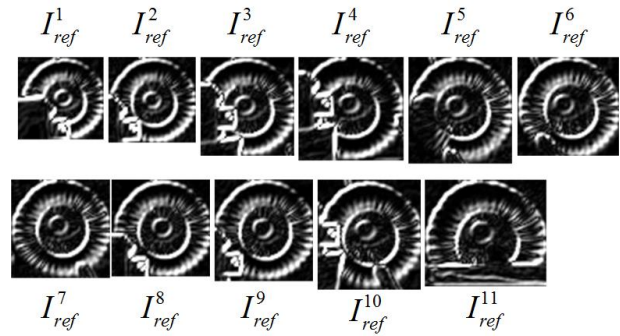
$$I_{ref}^1 \quad I_{ref}^2 \quad I_{ref}^3 \quad I_{ref}^4 \quad I_{ref}^5 \quad I_{ref}^6$$

$$I_{ref}^7 \quad I_{ref}^8 \quad I_{ref}^9 \quad I_{ref}^{10} \quad I_{ref}^{11}$$

Figure 9. Edge templates at air scene 1 in the method of Carol Martinez et al
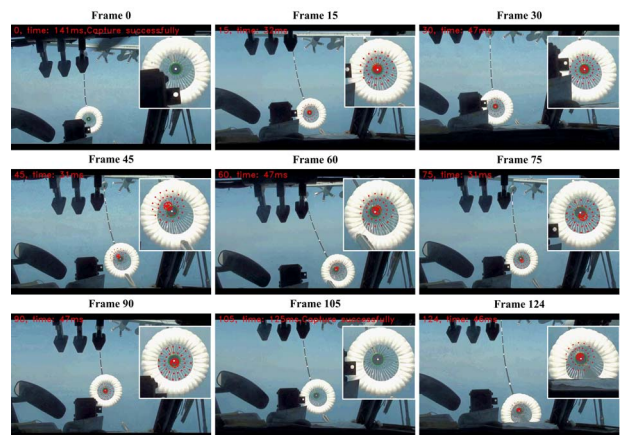

Figure 10. Nine result frames at air scene 1 in our method
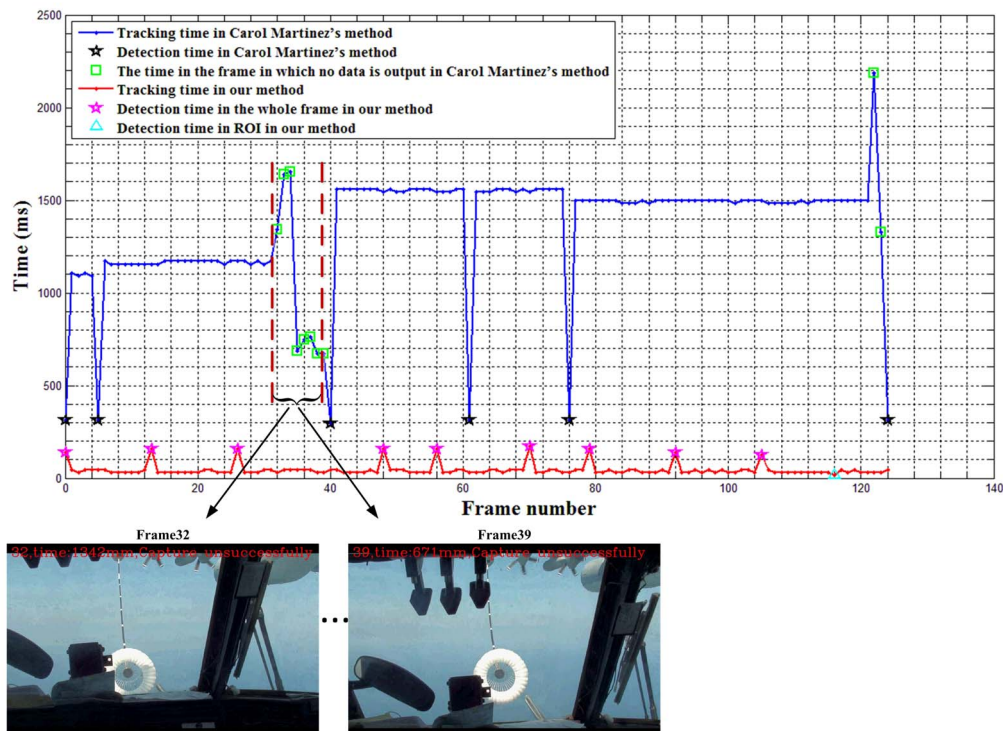

Figure 11. The processing time at the air scene 1

6.1 At Air Scene 1

125 frames of images with 1440×900 pixel size were used in the experiment at air scene 1. The experimental data were obtained from the website http://www.youtube.com/watch?v=nWmFpLVl8MQ. Eleven edge templates as shown in Figure 9 were used to find the drogue in the lowest resolution image in the image pyramid, the number of pyramid levels was pL=3 and the threshold used to segment the image was 85 in Carol Martinez's method [7]. In our method, the number of pyramid levels in the application was pL=3, the lowest threshold was T_0=20, the number of thresholds was k=53 and the increment of the threshold was ΔT=3 in equation (4), The aspect ratio was T_{Ratio}=0.7, the lower bound of the length of the minimum enclosing rectangle was T_{L1}=4 and the upper bound of the length of the minimum enclosing rectangle T_{L2} was equal to one third of the image's height in (6). The lower bound of the width of the minimum enclosing rectangle was T_{W1}=4 and the upper bound of the width of the minimum enclosing rectangle T_{W2} was equal to one third of the image's width in (7) and (8). The threshold for eliminating unnecessary contours was T_d=1.5 in (12); the number of the half-lines corresponding to each contour was 20, the length of each half-line was 60, the height of the small rectangle was three and the width of the small rectangle was eight in Section 3.2.1, The threshold of the vector angles was θ_T =80° in (18); the threshold T_W was 15 in (20); the number of samples was n_s=5 and the threshold T_r was six in the pseudo-code in Section 3.2.3, The number of interior particles was N_{in}=20, the number of exterior particles was N_{ex}=45, the radius of the circle C_{cir} was 15 and the parameter l was 15. The distance between two adjacent exterior state transition particles in the same half-line was Δs=22. Three exterior particles were in the same half-line and the angle between two adjacent half-lines was θ=24° in 4.1. The threshold T_P was 15 in 4.2; the threshold T_l was 15 in Section 5.

Nine result frames in our method are shown in Figure 10. The magnified target is displayed in the top-right corner of the frame in Figure 10. The green circle segmented in detecting and tracking process is the inner dark part of the drogue, the white point is the green circle's centre and the red points are the state transition particles in Figure 10. The comparison of processing time of each frame between our method and the method of Carol Martinez et al. is shown in Figure 11. The processing time of each frame in our method is clearly less than the processing time in the method of Carol Martinez et al. The processing time in our method is not affected by the scale of the drogue in the image. In Figure 11, the cyan triangle is the detection time (about 16ms) in ROI in the 116th frame and the nine magenta pentagrams stand for the detection time (less than 172ms) in the whole images in our method. The green squares represent the time in

which no data is output (or the algorithm thinks there is no target in the frame) in Carol Martinez's method. As shown in Figure 11, no data is output from the 32th frame to the 39th frame, though the targets in these frames are clear in Carol Martinez's method, while our method gives the drogue's positions in the entire frames. In the method of Carol Martinez et al., the processing time is affected by the size of the reference image. The detecting algorithm finds the first reference image with pixel size 213×206 in the 0th frame as shown in Figure 11 and the average tracking time corresponding to the first reference image is 1099.5*ms*. The detecting algorithm finds the second reference image with pixel size 217×213 in the fifth frame as shown in Figure 11 and the average tracking time corresponding to the second reference image is 1163.3*ms*. The detecting algorithm finds the third reference image with pixel size 246×254 in the 40th frame as shown in Figure 11 and the average tracking time corresponding to the second reference image is 1555.4*ms*. The speed indicators in our method are better than the method of Carol Martinez et al. as shown in Table 1. The percentage of correct location in the method of Carol Martinez et al. is less than 30%; in our method it is 100% when the location error threshold is five pixels, as shown in Figure 18.

6.2 At Air Scene 2

One hundred and ninety frames of images with pixel size 1440×900 were used in the experiment at air scene 2. The experimental data were obtained from the website http://www.youtube.com/watch?v=cG6rMZF6mIw. The appearance of the drogue at air scene 2 was different from the appearance of the drogue at air scene 1, but both had the two important features. Eleven edge templates were used to find the drogue as shown in Figure 12 and the threshold used to segment the image was 40 in Carol Martinez's method. The parameters in our method are the same as the parameters in Section 6.1. Nine result frames are shown in Figure 13 and the processing time of each frame is shown in Figure 14 in our method. The processing time of each frame in our method is clearly less than the processing time in the method of Carol Martinez et al. as shown in Figure 14.
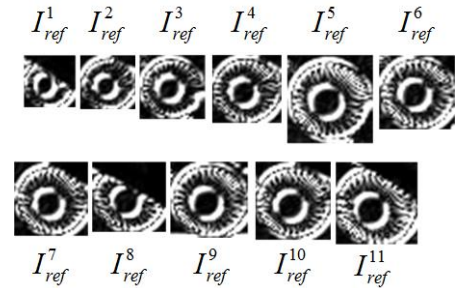

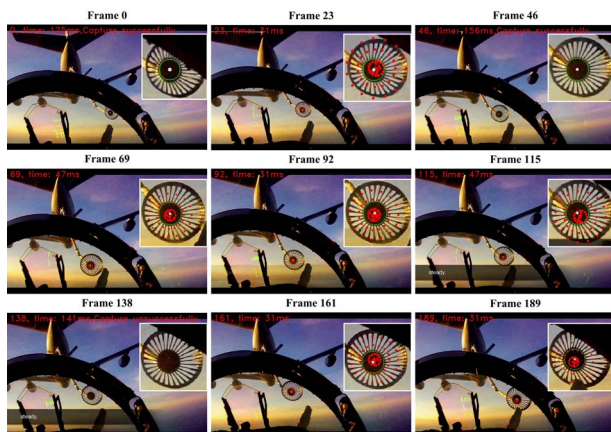**Figure 12.** Edge templates at air scene 2 in the method of Carol Martinez et al

Figure 13. Nine result frames at air scene 2 in our method

In Figure 14, the seven cyan triangles are the detection time in ROI and the 11 magenta pentagrams stand for the detection time in the whole frames in our method. The detection time represented as the black pentagrams in Figure 14 in Carol Martinez's method is obviously greater than the detection time in our method. The green squares are the time in which no data is output in Carol Martinez's method. As shown in Figure 14, no data is output from the 121th frame to the 129th frame and a wrong target is detected and tracked from the 130th frame to the 136th frame. This is probably because the drogue is partially occluded. In our method, only in the 138th frame is no data output. The speed indicators in our method are better than the method of Carol Martinez et al., as shown in Table 1. The percentage of correct

location in the method of Carol Martinez et al. is less than 20%, while in our method it is near 80% when the location error threshold is five pixels, as shown in Figure 18.

6.3 At Air Scene 3

123 frames of images with 1440×900 pixel size were used in the experiment at air scene 3. The experimental data were obtained from the website http://www.jokeroo.com/videos/cool/aerial-refueling.html. The appearance of the drogue at air scene 3 was different from the appearance of the drogues at air scene 1 and air scene 2, but all of them had two important features. Eleven edge templates were used to find the drogue as shown in Figure 15 and the threshold used to segment the image was 50 in Carol Martinez's method. The parameters in our method are same to the parameters in Section 6.1. Nine result frames are shown in Figure 16 and the processing time of each frame in our method is shown in Figure 17. The processing time of each frame in our method is clearly less than the processing time in the method of Carol Martinez et al. as shown in Figure 17 since the HMPMR-ICIA [21] algorithm adopted by Carol Martinez is a time-consuming iterative optimization method during tracking. The speed indicators in our method are better than the method of Carol Martinez et al., as shown in Table 1. The percentage of correct location in the method of Carol Martinez et al. is less than 70%, while in our method it is more than 80% when location error threshold is five pixels, as shown in Figure 18.
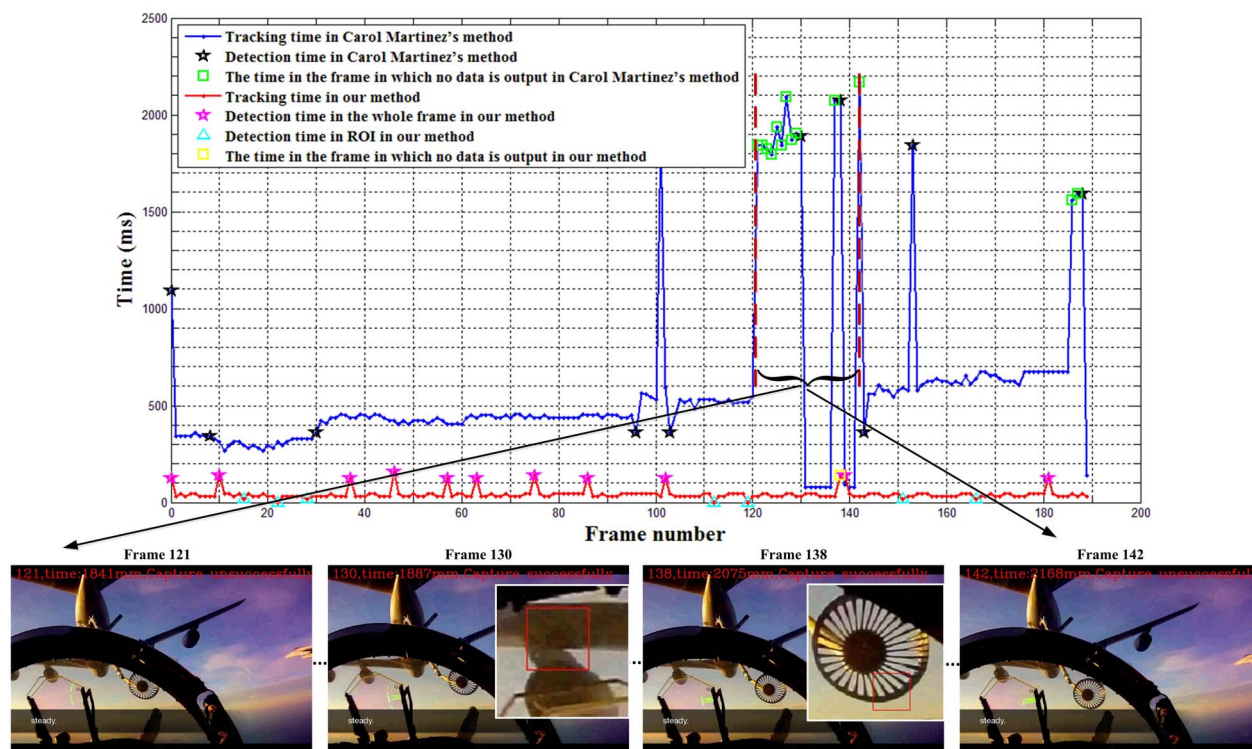

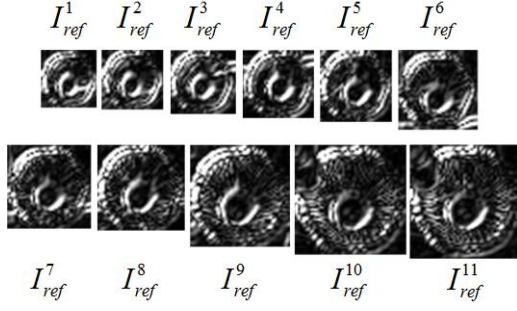**Figure 14.** The processing time at air scene 2

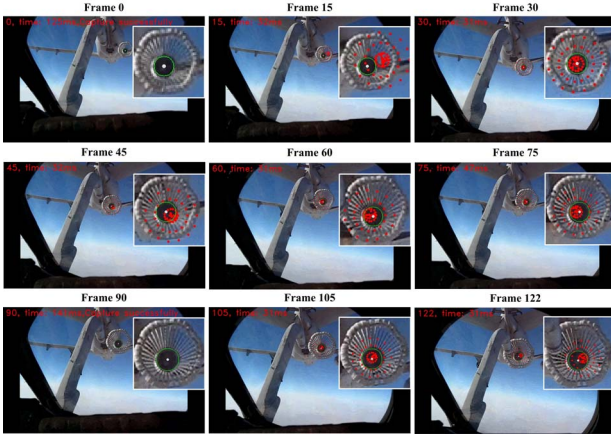**Figure 15.** Edge templates at air scene 3 in the method of Carol Martinez et al
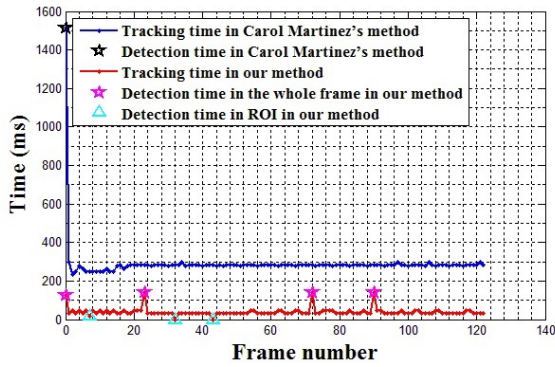

Figure 16. Nine result frames at air scene 3 in our method

Figure 17. The processing time of each frame at air scene 3

Scene and Method		t_{fave}(ms)	t_{max}(ms)	t_{min}(ms)	t_{ave}(ms)
Scene 1	Our method	45.3	172	16	45.3
	Carol Martinez	1346.2	9346	297	1362.1
Scene 2	Our method	42.9	282	≤1	43.1
	Carol Martinez	588.5	18828	78	631.7
Scene 3	Our method	38.1	141	≤1	38.1
	Carol Martinez	288.4	1513	234	288.4

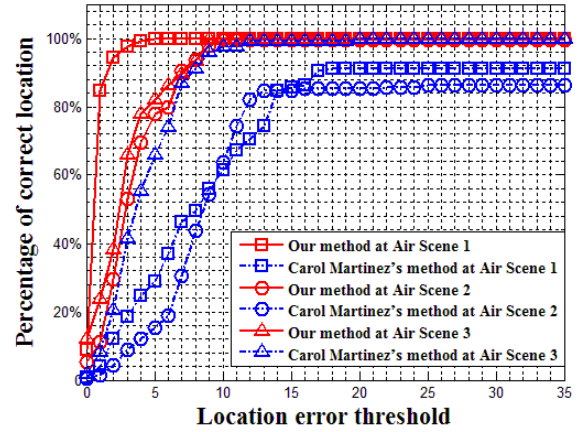**Table 1.** Comparison of processing time

Figure 18. Percentages of correct location at different location error thresholds

7. Conclusions

Detecting and tracking strategies for aerial refuelling tasks based on monocular vision were proposed. According to the drogue's prior knowledge, multi-threshold segmentation and shape-distinguishing arithmetic were used to detect the drogue. Multi-threshold segmentation decreased the rate of missed detection and the figure-distinguishing arithmetic pinpointed the drogue's position precisely. A new particle filter algorithm based on the drogue's shape was proposed to track the drogue. In the new particle algorithm, unique principles of state transition are defined to ensure tracking robustness even when the drogue's position or size change significantly in two adjacent frames. A strategy of switching between detection and tracking was proposed and the switching strategy enhanced robustness of our method. In the method, the inner dark part of the drogue was segmented precisely in the detecting and tracking process and the segmented circular part can be used to measure the spatial position of the drogue. The speed of the proposed method is fast, is less affected by the size of the drogue in the image and it is highly accurate. In the future, we will try to use the segmented circular part to measure the position of the drogue in Cartesian space based on monocular vision by the method of combing the 3D model of the drogue and the imaging principle.

8. Acknowledgments

Yingjie Yin, De Xu, Xingang Wang and Mingran Bai: Detection and Tracking Strategies for Autonomous Aerial Refuelling Tasks Based on Monocular Vision

11

9. References

[1] Speer T. E. (2008) Systems and methods for automatically and semiautomatically controlling aircraft refueling, The Boeing Company, US Patent. 7469863.

[2] Dellaquila R.V., Campa G., Napolitano M. R. et al. (2007) Real-time machine-vision-based position sensing system for UAV aerial refueling, Journal of Real-Time Image Processing, Vol. 1, No. 3, 213-224.

[3] Fravolini M. L., Mammarella M., Campa G. et al. (2010) Machine vision algorithms for autonomous aerial refueling for UAVs using the USAF refueling boom method, Studies in Computational Intelligence, Vol. 304, 95-138.

[4] Mammarella M., Campa G., Napolitano M.R. et al. (2008) Machine vision/GPS integration using EKF for the UAV aerial refueling problem, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 38, No. 6, 791-801.

[5] Williamson W. R., Glenn G. J., Dang V.T. et al. (2009) Sensor fusion applied to autonomous aerial refueling, Journal of Guidance, Control, and Dynamics, Vol. 32, No. 1, 262-275.

[6] Valasek J., Gunnam K., Kimmett J. et al. (2005) Vision-based sensor and navigation system for autonomous air refueling, Journal of Guidance, Control, and Dynamics, Vol. 28, No. 5, 979-989.

[7] Martinez C., Richardson T., Thomas P., et al. (2013) A vision-based strategy for autonomous aerial refueling tasks, Robotics and Autonomous Systems, Vol. 61, No. 8, 876-895.

[8] Ro K., Kamman J. W. (2010) Modeling and simulation of hose-paradrogue aerial refueling systems, Journal of Guidance, Control, and Dynamics, Vol. 33, No. 1, 53-63.

[9] Zhu Z. H., Meguid S. A. (2007) Modeling and simulation of aerial refueling by finite element method, International Journal of Solids and Structures, Vol. 44, No. 24, 8057-8073.

[10] Wang Y., Tang X., Cui Q. (2012) Dynamic appearance model for particle filter based visual tracking, Pattern Recognition, Vol. 45, No. 12, 4510-4523.

[11] Zhang X., Liu H., Sun X. (2013) Object tracking with an evolutionary particle filter based on self-adaptive multi-features fusion, International Journal of Advanced Robotic Systems, Vol. 10, No. 61, 1-11.

[12] Ponsa D., Lopez A. M. (2009) Variance reduction techniques in particle-based visual contour tracking, Pattern Recognition, Vol. 42, No. 11, 2372-2391.

[13] Hotta K. (2009) Adaptive weighting of local classifiers by particle filters for robust tracking, Pattern Recognition, Vol. 42, No. 5, 619-628.

[14] Kimmett J., Valasek J., Junkins J. L. (2002) Autonomous aerial refueling utilizing a vision based navigation system, in AIAA Guidance, Navigation, and Control Conference and Exhibit.

[15] Kimmett J., Valasek J., Junkins J.L. (2002) Vision based controller for autonomous aerial refueling, in Proceedings of the 2002 IEEE International Conference on Control Applications.

[16] Tandale M. D., Bowers R., Valasek J. L. (2005) Robust trajectory tracking controller for vision based probe and drogue autonomous aerial refueling, in AIAA Guidance, Navigation, and Control Conference and Exhibit.

[17] Junkins J. L., Schaub H., Hughes D. (2001) Noncontact position and orientation measurement system and method, US Patent 6366142.

[18] Fravolini M. L., Ficola A., Campa G. et al. (2004) Modeling and control issues for autonomous aerial refueling for UAVs using a probe-drogue refueling system, Aerospace Science and Technology, Vol. 8, No. 7, 611-618.

[19] Pollini L., Mati R., Innocenti M. (2004) Experimental evaluation of vision algorithms for formation flight and aerial refueling, in AIAA Modeling and Simulation Technology Conference and Exhibit.

[20] Lu C. P., Hager G. D., Mjolsness E. (2010) Fast and globally convergent pose estimation from video images, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 6, 610-622.

[21] Ishikawa T., Matthews I., Baker S. (2002) Efficient image alignment with outlier rejection, Technical Report CUM-RI-TR-02-27, Carnegie Mellon University Robotics Institute.

[22] Martinez C., Mejias L., Campoy P. (2011) A multi-resolution image alignment technique based on direct methods for pose estimation of aerial vehicles, in 2011 International Conference on Digital Image Computing: Techniques and Applications.

[23] Kong B., Wang Z., Tan Y. (2012) Algorithm of laser spot detection based on circle fitting, Infrared and Laser Engineering, Vol. 31, No. 3, 275-279.

[24] Nister D. (2005) Preemptive RANSAC for live structure and motion estimation, Machine Vision and Application, Vol. 16, No. 5, 321-329.

[25] Doucet A., Godsill S., Andrieu C. (2000) On sequential Monte Carlo sampling methods for Bayesian filtering, Statistics and Computing, Vol. 10, No. 3, 197-208.