Chapter 6. Vision-Based Relative Position Estimation

Abstract

Probe-and-drogue refueling is widely used owing to its simple requirement for refueling equipment and flexibility. For autonomous aerial refueling, determining the distance between an unmanned receiver aircraft and a tanker aircraft is of great importance. In this chapter, a vision-based method is proposed to estimate the position of the drogue by using a camera. This method is a two-step process. The first step is to detect the markers fixed on the drogue and match them as a circle. The second step is to improve the image robustness. In addition, the proposed method is verified in the simulation with a virtual reality toolbox. Simulation results indicate that the proposed method can track the circle steadily and estimate its position in real time.

I. Introduction

Along with the development of the UAV technology, Autonomous Aerial Refueling (AAR) systems are urgently needed. However, the probe-and-drogue system has an apparent drawback, which is susceptible to disturbances, making docking very difficult [1]. Thus, the autonomous aerial refueling requires precise relative position between the receiver aircraft and the drogue of the refueling system [2].

During the past decades, researchers have been making significant efforts to design position estimation methods, including Inertial Measurement Unit (IMU), Global Positioning System (GPS) [3], and vision-based position estimation method [4]. One aspect to note is that, due to the restriction of the safety, it is not permitted to attach electronic devices onto the drogue. Relative position information can be obtained from IMU measurements, but zero drift and accumulative error result in its accuracy not meeting the requirements. The GPS method has been made in 5cm to 10cm accuracy for formation flying, but problems emerge with accuracy decreasing because of signal blocked or other interference factors. In addition, it is hard to attach the GPS equipment to the drogue. Thus, as a newly-developed contactless method, the vision-based position sensor like a camera is a preferable solution to get the relative position [5].

The research on vision-based position estimation has been developing in the world, and many meaningful achievements have been made [6–8]. The existing schemes of vision-based refueling systems can be classified into two groups: image-based algorithm and feature tracking algorithm. The image-based algorithm regards the visual sensor as a two-dimensional sensor, whose characteristics such as image Jacobian matrix and gray value can be integrated into the control law. A typical example of this kind of method is the algorithm using the predictive image for vision aids [7]. The feature tracking algorithm is to obtain the relative position by means of acquiring and tracking specific features (points, lines, etc.) from a visual sensor. Typical examples of this kind of method include visual positioning systems based on infrared vision sensors [9] and VisNAV active vision navigation systems [10, 11].

In this chapter, the main algorithm is a kind of feature tracking algorithm. For example, in the VisNAV system, the position and attitude information is obtained by LHM [12] algorithm which is based on a monocular camera and some infrared Light Emitting Diode (LED) marking points. Nevertheless, the LHM algorithm is an iterative algorithm, which is somewhat time-consuming. In face of such a situation, in this chapter, a simpler feature point detecting and matching method with relatively high efficiency and reliability is proposed. In addition, some extra measures are also taken to improve the robustness of the system.

The main features of this chapter are as follows.

1) A feature point algorithm of detecting and matching the markers as a circle is proposed.
2) In order to improve the robustness of the system, a Kalman filter (KF) based method to reduce observation errors is proposed. In addition, several general correspondence methods are proposed to reduce the influence of noise, redundant and losing points.
3) Simulations are carried out to validate the effectiveness of the proposed methods.

This chapter is organized as follows. Some preliminaries and problem formulation are introduced in Section II. In Section III, the main algorithms used in this chapter are presented. Then, in Section IV, the details and results of the simulations are expressed. Finally, in Section V, the conclusions are presented.

II. Preliminaries and Problem Formulation

A. The Layout of Markers

In order to determine the distance among the receiver aircraft, the tanker aircraft and the drogue of the refueling system, it is necessary to place some markers on the surface of the latter two. Moreover, in order to represent the geometric characteristics of the drogue, the markers (see Fig. 1) can be distributed on the circle of the drogue canopy with different intervals between them. Combined with physical and algorithmic filtering methods, markers can be easily extracted from the image.

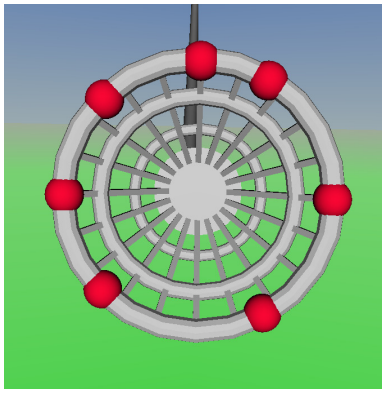

Fig. 1: Markers located at the drogue

B. Coordinate System Transformation

In this chapter, the coordinate systems are defined as follows (see Fig. 2). The camera coordinate system $o_{\mathrm{c}} - x_{\mathrm{c}}y_{\mathrm{c}}z_{\mathrm{c}}$ is attached to the camera. Its origin is the optical center of the camera, with x_{c} axis pointing forward, y_{c} axis pointing right, z_{c} axis pointing downward. The other coordinate system is the drogue coordinate system $o_{\mathrm{d}} - x_{\mathrm{d}}y_{\mathrm{d}}z_{\mathrm{d}}$, whose origin is the center of the drogue. Moreover, its orientation is the same as the camera coordinate system.

Assume that vectors $\mathbf{p}_{\mathrm{c}} \triangleq [x_{\mathrm{c}}\ y_{\mathrm{c}}\ z_{\mathrm{c}}]^{\mathrm{T}}$ and $\mathbf{p}_{\mathrm{d}} \triangleq [x_{\mathrm{d}}\ y_{\mathrm{d}}\ z_{\mathrm{d}}]^{\mathrm{T}}$ are in two coordinate systems above, which satisfy [13]:

$$\mathbf{p}_{\mathrm{c}} = \mathbf{R}_{\mathrm{d}}^{\mathrm{c}}\mathbf{p}_{\mathrm{d}} + \mathbf{t}_{\mathrm{d}}^{\mathrm{c}} \tag{1}$$

where $\mathbf{R}_{\mathrm{d}}^{\mathrm{c}} \in \mathbb{R}^{3\times3}$ is the rotation matrix, and $\mathbf{t}_{\mathrm{d}}^{\mathrm{c}} \in \mathbb{R}^{3}$ is the translation vector. The rotation matrix $\mathbf{R}_{\mathrm{d}}^{\mathrm{c}}$ from the drogue coordinate system to the camera coordinate system can be shown as

$$\mathbf{R}_{\mathrm{d}}^{\mathrm{c}} = \mathbf{R}_{\mathrm{z}}^{\mathrm{T}}(\psi)\mathbf{R}_{\mathrm{y}}^{\mathrm{T}}(\theta)\mathbf{R}_{\mathrm{x}}^{\mathrm{T}}(\phi). \tag{2}$$

In the equation above, with three principal axes, a rotation of angle ϕ about the x-axis is defined as

$$\mathbf{R}_{\mathrm{x}}(\phi) \triangleq \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}. \tag{3}$$

Similarly, a rotation of angle θ about the y-axis is defined as

$$\mathbf{R}_{\mathrm{y}}(\theta) \triangleq \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}. \tag{4}$$

Besides, a rotation of angle ψ about the z-axis is defined as

$$\mathbf{R}_{\mathrm{z}}(\psi) \triangleq \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where ψ, θ and ϕ are the Euler angles.

In the process of aerial refueling, the rotation between the receiver aircraft and the drogue is limited in a small range to ensure the safety, which can be ignored. Thus, assume that there is only translation which can be expressed as

$$\mathbf{p}_{\mathrm{c}} = \mathbf{p}_{\mathrm{d}} + \mathbf{t}_{\mathrm{d}}^{\mathrm{c}}. \tag{6}$$

Using (6) , equations of position parameters can be established and solved.

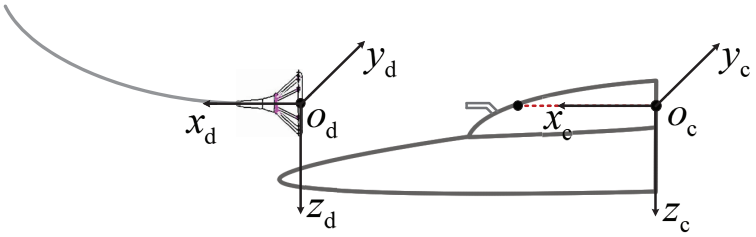

Fig. 2: The coordinate systems of the refueling system [14]

C. Camera Pinhole Model

Assume that a vector $\mathbf{p}_\mathrm{i} \triangleq [u\ v]^\mathrm{T}$ is in the image coordinate system $o_\mathrm{i} - x_\mathrm{i}y_\mathrm{i}$. The camera pinhole model (see Fig. 3) is used to transform $\mathbf{p}_\mathrm{c}$ and $\mathbf{p}_\mathrm{d}$ to $\mathbf{p}_\mathrm{i}$ as follows

$$l\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_\mathrm{x} & 0 & u_0 & 0 \\ 0 & \alpha_\mathrm{y} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_\mathrm{c} \\ y_\mathrm{c} \\ z_\mathrm{c} \\ 1 \end{bmatrix} = \mathbf{M} \begin{bmatrix} \mathbf{R}_\mathrm{d}^\mathrm{c} & \mathbf{t}_\mathrm{d}^\mathrm{c} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_\mathrm{d} \\ y_\mathrm{d} \\ z_\mathrm{d} \\ 1 \end{bmatrix}, \tag{7}$$

and

$$\mathbf{M} = \begin{bmatrix} \alpha_\mathrm{x} & 0 & u_0 \\ 0 & \alpha_\mathrm{y} & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{8}$$

where l in (7) is the scaling factor; $\mathbf{M}$ is the camera intrinsic matrix, in which α_x, α_y, u_0 and v_0 are determined by camera calibration [14].

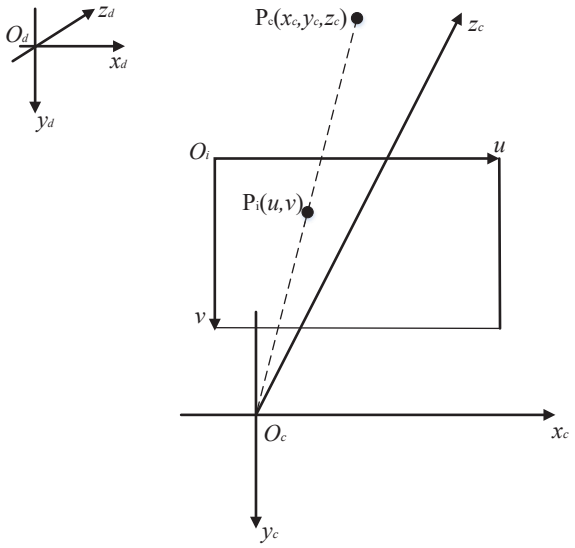

Fig. 3: Camera pinhole model [15]

D. Problem Formulation

According to the descriptions above, it is obtained that some markers are placed on the drogue. Let the number of the markers be N. As all markers are placed in the same plane, their depth information is identical, which can be expressed as s. Thus the markers in the image can be described as $(u_i, v_i, s),\ i = 1, 2, \cdots, N$. Since the markers compose a circle, it is necessary to obtain the center coordinate and radius of the circle, which can be expressed as (a, b, r) with (a, b) denoting the center coordinate of the circle, r the radius of the circle. With these parameters, the relative distance between the drogue and the receiver aircraft is obtained. However, since the external environment is complicated and full of interferences, some essential measures should be taken, which can ensure the accuracy of the parameters and improve the robustness of the whole system.

There are two major tasks in this chapter: drogue recognition and enhancing detection robustness. Enhancing detection robustness is based on the prediction of the markers' coordinates, and it can be divided into three parts corresponding to different situations. Therefore, for simplicity, they are formulated into the following four steps.

1) Step 1: Assume that there is no disturbance and the whole refueling system works well. According to the known parameters $(u_i, v_i, s),\ i = 1, 2, \cdots, N$, get the the center and radius of the circle (a, b, r).
2) Step 2: According to the relative distance between the drogue and the receiver aircraft got in *Step 1*, estimate the coordinates of the markers at the next moment and revise the current data.
3) Step 3: Assume that there are some noise and redundant points in the image. According to the known parameters $(u_i, v_i, s),\ i = 1, 2, \cdots, N$ and the equation of the circle, eliminate the influence of interferences.
4) Step 4: Assume that there are some markers undetectable. According to the current and predictive coordinates of the markers, bring forward corresponding measures.

Next, the main algorithm will be introduced in Section III in detail.

III. Main Algorithm for Position Estimation

A. Markers Detecting and Matching

As shown above, markers placed on the drogue are arranged as a circle. In the process of aerial refueling, the receiver aircraft should track the tanker aircraft with high accuracy. Thus, the angle of attack of the receiver aircraft is very small, according to which the drogue can be approximated as a circle. Its projection on the image can be expressed in the function form as

$$(x - a)^2 + (y - b)^2 = r^2 \tag{9}$$

Let N be the number of all detected markers. The detected markers can be expressed as $\mathbf{p}_i \triangleq [x_i\ y_i]^{\mathrm{T}},\ i = 1, 2, \cdots, N$. In order to describe the difference between the observed value and the estimated value, the residuals ε_i is used, which can be shown as

$$\varepsilon_i = (x_i - a)^2 + (y_i - b)^2 - r^2. \tag{10}$$

The cost function can be the sum of the residuals' square, that is

$$J = \sum_{i=1}^{N} \varepsilon_i^2 = \sum_{i=1}^{N} \left[(x_i - a)^2 + (y_i - b)^2 - r^2 \right]^2. \tag{11}$$

According to the principle of the least square, when the partial derivative of the cost function equals zero, the optimal fitting result is achieved, which can be written in the numerical form as

$$\begin{cases} \dfrac{\partial J}{\partial a}=-4(x_i-a)\sum\limits_{i=1}^{N}\left[(x_i-a)^2+(y_i-b)^2-r^2\right]=0 \\ \dfrac{\partial J}{\partial b}=-4(y_i-b)\sum\limits_{i=1}^{N}\left[(x_i-a)^2+(y_i-b)^2-r^2\right]=0 \\ \dfrac{\partial J}{\partial r}=-4r\sum\limits_{i=1}^{N}\left[(x_i-a)^2+(y_i-b)^2-r^2\right]=0. \end{cases} \tag{12}$$

With these functions, the parameters of the circle are obtained as

$$\begin{cases} a=\dfrac{(\overline{x^2}\cdot\overline{x}+\overline{x}\cdot\overline{y^2}-\overline{x^3}-\overline{xy^2})(\overline{y}^2-\overline{y^2})}{2(\overline{x}^2-\overline{x^2})(\overline{y}^2-\overline{y^2})-2(\overline{x}\cdot\overline{y}-\overline{xy})^2}- \\ \qquad\dfrac{(\overline{x^2}\cdot\overline{y}+\overline{y}\cdot\overline{y^2}-\overline{x^2y}-\overline{y^3})(\overline{x}\cdot\overline{y}-\overline{xy})}{2(\overline{x}^2-\overline{x^2})(\overline{y}^2-\overline{y^2})-2(\overline{x}\cdot\overline{y}-\overline{xy})^2} \\ b=\dfrac{(\overline{x^2}\cdot\overline{y}+\overline{y}\cdot\overline{y^2}-\overline{x^2y}-\overline{y^3})(\overline{x}^2-\overline{x^2})}{2(\overline{x}^2-\overline{x^2})(\overline{y}^2-\overline{y^2})-2(\overline{x}\cdot\overline{y}-\overline{xy})^2}- \\ \qquad\dfrac{(\overline{x^2}\cdot\overline{x}+\overline{x}\cdot\overline{y^2}-\overline{x^3}-\overline{xy^2})(\overline{x}\cdot\overline{y}-\overline{xy})}{2(\overline{x}^2-\overline{x^2})(\overline{y}^2-\overline{y^2})-2(\overline{x}\cdot\overline{y}-\overline{xy})^2} \\ r=\sqrt{a^2-2\overline{x}a+b^2-2\overline{y}b+\overline{x^2}+\overline{y^2}} \end{cases} \tag{13}$$

where $\overline{x}$ and $\overline{y}$ denote the average values of x and y, and $\overline{x^m y^n}=\dfrac{\sum_{i=1}^{N}x_i^m y_i^n}{N}, m,n\in[0,3]$.

On the basis of camera pinhole model, the depth s of the plane at which the markers are located can be obtained by the radius of the circle. The function is as follows

$$\frac{R_{\mathrm{dr}}}{s+f}=\frac{r}{f} \tag{14}$$

where R_{dr} is the actual radius, f is the focal length of the camera. Let the coordinate of the probe in the image be $\mathbf{p}_{\mathrm{i}}\triangleq[u\ v]^{\mathrm{T}}$, and the distance between the probe and the camera in z-axis be d. The relative distance between the drogue and the probe can be expressed as

$$\begin{cases} \Delta x=|a-u|\cdot\dfrac{R_{\mathrm{dr}}}{r} \\ \Delta y=|b-v|\cdot\dfrac{R_{\mathrm{dr}}}{r} \\ \Delta z=\left|\dfrac{fR_{\mathrm{dr}}}{r}-f-d\right|. \end{cases} \tag{15}$$

Although the form of this algorithm is complex, its time complexity is just $O(n)$, which is suitable for computer implementation.

B. KF and Robust Image Tracking Algorithm

In this subsection, KF is applied to estimate the position of markers, which can reduce the influence of error points and improve the matching accuracy. The equations of state and measurement are given as follows

$$\mathbf{x}(k+1) = \mathbf{\Phi}(k+1,k)\mathbf{x}(k) + \mathbf{\Gamma}(k+1,k)\mathbf{W}(k) \tag{16}$$

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{V}(k). \tag{17}$$

In above functions, $\mathbf{x}(k) \in \mathbb{R}^6$ is the state vector of the system state, which is written as

$$\mathbf{x} = \begin{bmatrix} \Delta x & \Delta y & \Delta z & \Delta v_{\mathrm{x}} & \Delta v_{\mathrm{y}} & \Delta v_{\mathrm{z}} \end{bmatrix}^{\mathrm{T}} \tag{18}$$

where $\Delta x, \Delta y, \Delta z$ are the relative distances between the receiver aircraft and the drogue along three axes, respectively. And $\Delta v_{\mathrm{x}}, \Delta v_{\mathrm{y}}, \Delta v_{\mathrm{z}}$ are velocity differences; $\mathbf{z}(k) \in \mathbb{R}^3$ is the observation vector, which is expressed as

$$\mathbf{z} = \begin{bmatrix} u & v & s \end{bmatrix}^{\mathrm{T}}. \tag{19}$$

and $\mathbf{H}(k)$ can be obtained from the correspondence of the image and the real world. Besides, other parameters can be expressed as

$$\mathbf{\Phi}(k+1,k) = \begin{bmatrix} \mathbf{0}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \tag{20}$$

$$\mathbf{\Gamma}(k+1,k) = \begin{bmatrix} \mathbf{0}_3 \\ \mathbf{I}_3 \end{bmatrix}. \tag{21}$$

Given the initial value, the well-known KF consisting of prediction and estimation parts can start to iterate. The estimation of the states can be sent to the autopilot for the docking control. KF can also improve the system robustness performance enormously which will ensure the normal operation of the whole refueling system.

C. The Dispose of Noise and Redundant Points

As the environment around the refueling equipment is complex, it is a common phenomenon that some noise and redundant points appear in the image, which may affect the matching of the drogue. These redundant points may emerge for the following reasons: direct sunlight, light reflecting on the surface of the tanker aircraft, the noise points generated by the camera. The corresponding measures are given as follows.

For the drogue of the refueling system, let the center coordinate of estimation in world space be $\mathbf{z} = [\Delta x\ \Delta y\ \Delta z]^{\mathrm{T}}$. According to the projection relation, the next center and radius of the circle in the image can be forecasted as (a, b) and r, and the estimation errors are Δa, Δb, Δr, respectively. For the image point (u, v) at the next moment, the judgment rules can be expressed as follows

$$\sqrt{(u_i - a)^2 + (v_i - b)^2} > \sqrt{\Delta a^2 + \Delta b^2} + r + \Delta r \tag{22}$$

or

$$\sqrt{(u_i - a)^2 + (v_i - b)^2} < r - \Delta r. \tag{23}$$

The two formulas have a similar effect, which can distinguish the unwanted points with high efficiency. If the point satisfies the condition (22) or (23), it will be removed.

In addition, as the markers located at the drogue are placed as a circle, let the current circle in the image be (a, b, r). If the fitting error is greater than the threshold value ε set earlier, this point should be removed. The function can be expressed as

$$\frac{(u_i - a)^2 + (v_i - b)^2 - r^2}{r} > \varepsilon. \tag{24}$$

In actual operation, the value of ε can be got from real experiments.

The first algorithm shown above removes the unsuitable points in the aspect of the markers' movement tendency, while the other does the same in the aspect of markers' geometry distribution property. Together with other necessary logical judgments, most errors can be discovered and got rid of, which can ensure the accuracy of position estimation.

D. The Algorithm of Losing Points

When the probe-and-drogue refueling system is operating, there is a common phenomenon that some of the markers may be out of the camera's view or obscured by the probe or other obstacles. Due to the determination of parameters of the circle (a, b, r), three detectable markers should be obtained at least. By estimating the coordinates of the markers in the image, whether or not there are markers out of sight can be determined in advance. Different states and necessary countermeasures are listed as follows.

State 1: If some of the markers are out of sight, and other visible markers are near the probe, this situation means that the invisible markers are obscured by the probe. If they are just out of sight for a short time, which is less than the threshold value t_{i} , the estimation results of KF can fill the missing data directly, which can maintain system normal operation. However, at the same time, an alarm is provided until all return to normal.

State 2: If some markers are blocked by the probe for such a long time that exceeds the threshold value t_{i} , it is necessary to check the number of markers available. If the number is less than three, it is illustrated that the position estimation system is in bad condition, which may lead to huge safety risks. Thus, the receiver aircraft should stop refueling process immediately. Until the receiver aircraft returns to a safe place, a next refueling attempt is allowed to begin.

State 3: If some markers are out of sight, and the existing markers are near the four boundaries of the camera's view, it means that the invisible markers are beyond the camera's view. If this situation occurs, it implies that there is something wrong with the relative position between the receiver aircraft and the tanker aircraft. If all the missing markers return to view after a short period, which is less than the threshold value t_{p}, the refueling process can be permitted to carry on. If not, the refueling process must be stopped at once. An alarm is also required in this situation.

The threshold values t_{i} and t_{p} can be got from real experiments. However, according to the intensity of the events, t_{p} is much smaller than t_{i}. With these countermeasures, the robustness of the whole refueling system can be highly improved.

IV. Deep Learning-Based Drogue Detection

The target-based detection algorithms are susceptible to light interference and sensitive to weather conditions. Object detection algorithms based on deep learning, on the other hand, can directly learn the features of objects from images, eliminating the need for target installation on drogue and providing strong robustness. Early object detection algorithms consisted of two steps, namely candidate box generation and candidate box classification and localization, referred to as two-stage algorithms, such as R-CNN[16], Faster-RCNN[17], and FPN[18]. In contrast to two-stage algorithms, one-stage algorithms merge the two steps into one, significantly improving detection speed. Examples of one-stage algorithms include YOLO[19], SSD[20], and RetinaNet[21], with the YOLO series being the most widely used at present. In this chapter, the YOLOv5 algorithm is employed for drogue detection.

A. Network Architecture

The network architecture of YOLOv5 is illustrated in Fig. 4 and can be divided into four components: Input, Backbone, Neck, and Output. A key feature of YOLOv5 is its ability to detect objects on three different scales of feature maps, obtained through downsampling, corresponding to 1/32, 1/16, and 1/8 of the input image dimensions. The largest feature map is responsible for detecting small objects. For each point on the feature map, predictions are made using three anchor boxes as priors.

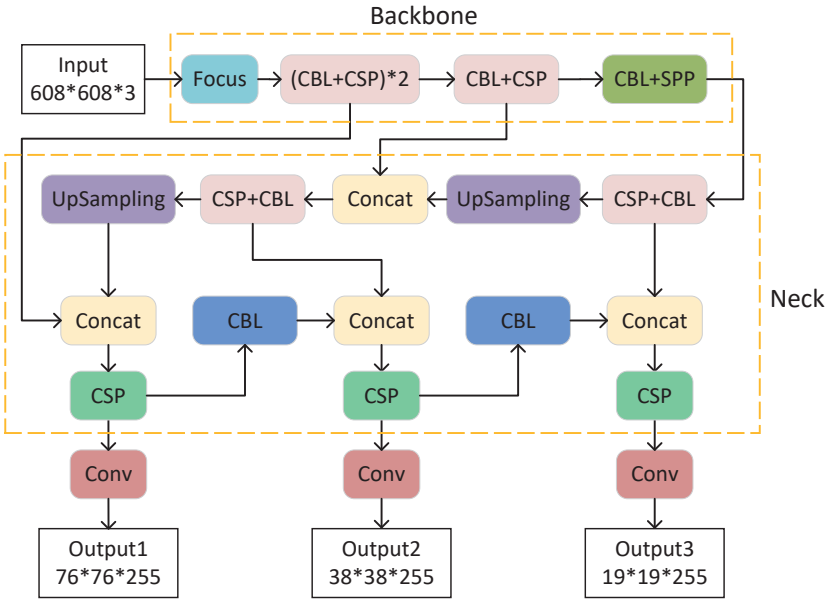

Fig. 4: The network architecture of YOLOv5

1) Input: (i) **Mosaic data augmentation**. The Mosaic data augmentation method was introduced in YOLOv4[22]. Its main idea is to concatenate four images together by randomly scaling, cropping, and arranging them. This concatenated image is then used as training data. The advantage of this approach is that it enhances the background diversity of the images and effectively increases the training batch size by combining four images into one.

(ii) **Adaptive anchor box computation**. In the algorithm, initial anchor boxes with predefined aspect ratios are set for different datasets. During the training process, the network computes predicted boxes based on these initial anchor boxes, compares them with the ground truth boxes, measures the differences between the two, and then updates the network parameters in a backward manner. YOLOv5 incorporates the functionality of adaptive computation of initial anchor boxes into the code, automatically calculating the optimal anchor box values for different training sets during each training iteration.

(iii) **Adaptive image scaling**. In the process of collecting datasets, it is common for images to have varying dimensions. A commonly used approach is to uniformly scale the images during data preprocessing to obtain a standardized size before inputting them into the network. Popular sizes used in the YOLO algorithm include 416×416 and 608×608. To achieve the standardized size, padding is applied to the edges of the images that fall short after scaling. In the prediction stage, YOLOv5 optimizes the padding algorithm, minimizing the extent of padding required and thereby improving the speed of object detection.

2) Backbone: (i) **Focus structure**. The most critical aspect of the Focus structure is the slicing operation. In YOLOv5, a $608 \times 608 \times 3$ image enters the Focus structure, undergoes the slicing operation, resulting in a $304 \times 304 \times 12$ feature map. Subsequently, a convolution operation with a kernel size of 32 is performed, transforming it into a $304 \times 304 \times 32$ feature map.

(ii) **CSP structure**. The CSP structure, inspired by CSPNet[23], addresses the issue of excessive computational complexity in inference from the perspective of network architecture design. YOLOv5 incorporates two types of CSP structures, with the CSP1_X structure applied to the Backbone and the CSP2_X structure utilized in the Neck.

3) Neck: The Neck network employs the FPN+PAN structure, drawing inspiration from PANet[24]. FPN is a top-down structure that utilizes upsampling to propagate and fuse high-level feature information, generating feature maps for prediction. YOLOv5 introduces a bottom-up feature pyramid after the FPN layer, which includes two RAN structures. By combining these two components, the FPN structure can transmit strong semantic features from top to bottom, while the feature pyramid is responsible for propagating strong localization features from bottom to top. The combination of these two components performs parameter aggregation operations on different detection layers from different backbone layers.

4) Output: (i) **Loss function**. The loss function for object detection tasks typically consists of two components: classification loss and bounding box regression loss. YOLOv5 adopts the CIoU Loss[25] as its loss function, which is derived from the DIoU Loss[26]. The CIoU Loss combines the considerations of overlapping area, center point distance, and aspect ratio of bounding boxes. It can be represented as

$$L_{\text{CIoU}} = 1 - \text{CIoU} = 1 - (\text{IoU} \ - \frac{\text{Distance\_2}^2}{\text{Distance\_C}^2} - \frac{v^2}{(1 - \text{IoU}) + v}) \tag{25}$$

where v is defined as a measure of aspect ratio consistency, plays a role in assessing the consistency of object's width and height ratios. It is defined as

$$v = \frac{4}{\pi^2}(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w^p}{h^p}). \tag{26}$$

(ii) **Non-Maximum suppression**. During the post-processing stage of object detection, multiple bounding boxes may be predicted for a single object, leading to overlapping detections. Non-Maximum Suppression (NMS) is commonly employed to identify the most relevant bounding box among them. Since CIoU incorporates the factor v, and there is no ground truth information available during prediction, DIoU is used instead. Firstly, the boxes for a specific class are sorted in descending order based on their confidence scores. Next, DIoU is computed, and boxes with DIoU values below a threshold are retained. Finally, the resulting output consists of the object's center position (u, v), width w, height h, confidence score p, and class probability $c_0, c_1, c_2, \ldots, c_{nc-1}$, as depicted in Fig. 5.

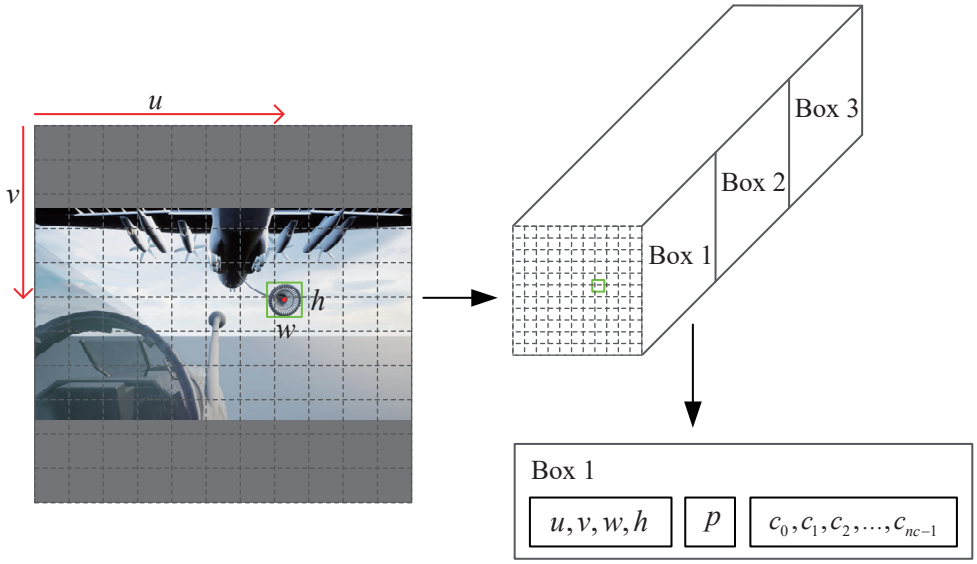

Fig. 5: The network architecture of YOLOv5

Among them, nc represents the number of categories, and the class probabilities are represented as a vector of length nc. The index corresponding to the maximum value in the vector indicates the predicted class. Since only the detection of a single object, namely "drogue" is performed, nc is set to 1.

B. Position esitimation

From the depth image, the depth value s of the drogue's center point coordinates (u, v) can be obtained. Using (7), the cone cover's coordinates p_c in the camera coordinate system can be derived, which subsequently leads to the calculation of p_d.

C. Dataset

In AARSim, the data collection process for the dataset is highly convenient. By adjusting the position of the tanker, images can be obtained at arbitrary distances and angles. Additionally, the camera resolution can be adjusted, with a resolution of 1280×720 set for this experiment.

TABLE I: The training parameter settings for the experiment

Parameters	Value
Model	YOLOv5s
Epochs	300
Image size	640×640
Batch size	16
Learning rate	0.01
Optimizer	SGD

D. Training approach

The experiment was conducted using an Intel(R) Core(TM) i7-10700F CPU, 32GB of RAM, and an NVIDIA RTX 2060 SUPER 8G GPU, on the Windows 11 operating system. PyCharm was utilized for development, and training, validation, and testing were performed with the same parameters. The training parameter settings for the experiment are presented in Table I.

V. Simulation and Results

A. Simulation Environment

In order to observe the simulation results intuitively, a three-dimensional (3D) simulation model is created by the virtual reality toolbox of Matlab, which emulates the process of aerial refueling precisely. The interface of this model is shown as follows (see Fig. 6).

The coordinate system of the virtual camera in this simulation model is different from the one in Section 2. Its origin is the current location set by the user, with x_{c} axis pointing forward, y_{c} pointing upward, z_{c} pointing right. In addition, the pixel resolution of the virtual camera is 860 pixels $\times$ 480 pixels. Moreover, its maximum frame rate is 50 frames per second. What is more, its angle of view is 90 degrees.

In order to test the effectiveness and practicability of the vision-based algorithm proposed in Section III, let the drogue stay still in a series of simulations, and at the same time, the receiver aircraft moves along the trajectory set in advance, for example, sinusoidal movement in three axes. According to the image acquired by the virtual camera on the receiver aircraft, the relative distance between the receiver aircraft and the drogue can be calculated.

B. Marker Identification

For the convenience of establishing the model, some colored spheres substitute for the passive reflectors are attached to the drogue and the tanker aircraft. The color of the spheres on the drogue is set to be red, while green on the tanker aircraft. The two kinds of spheres are distinguished by color, and then treated differently.

In the image obtained by the virtual camera, the markers appear to be bright. Thus, image gray processing (27) and thresholding function (28) are sufficient to detect markers. Then erode the bright pixel blob to eliminate noise according to the threshold value w. Finally, the center coordinates of the pixel blob can be obtained, namely, pixel coordinates of the markers.

$$\mathbf{Gray}(u,v) = 2\mathbf{R}(u,v) - \mathbf{G}(u,v) - \mathbf{B}(u,v) \tag{27}$$

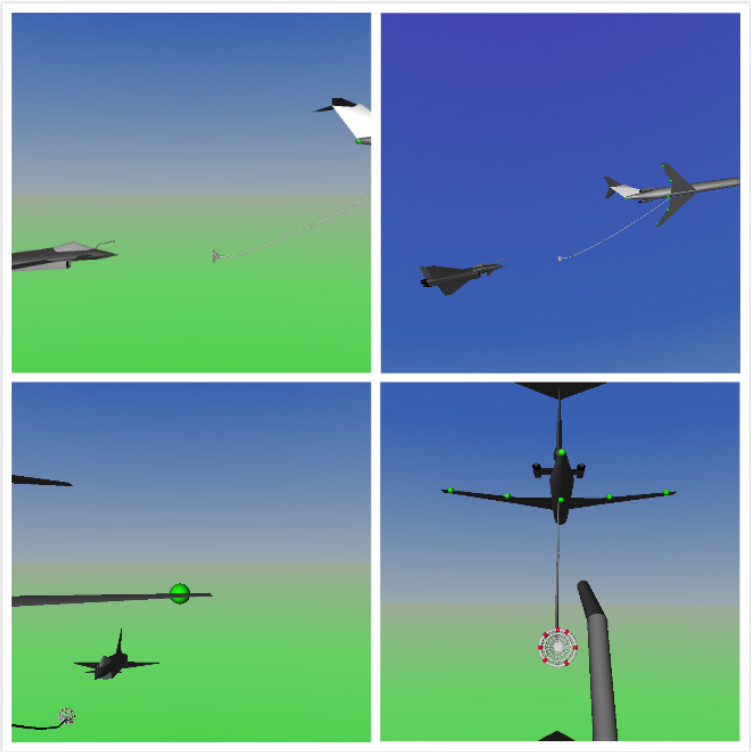

Fig. 6: Aerial refueling of VR simulation.

$$\mathbf{Gray}(u,v) = \begin{cases} 255, \ if \ \mathbf{Gray}(u,v) \geq w \\ 0, \ otherwise \end{cases} . \tag{28}$$

C. Simulation Results

The whole simulation lasts for 200 seconds, while real-time pictures are displayed in two windows. The images of different situations are listed as follows (see Figs. 7 and 8)

From these figures, it is obtained that the circle matching algorithm can work well even if some markers are blocked or beyond the boundary. Besides, even though the external environment is complex, the system can ensure a high-precision position estimation. These results show the high robustness of this position estimation system. The positioning data and actual data are compared as follows (see Fig. 9)

In Fig. 9, the dotted line represents the relative distance solved by vision-based position estimation algorithm, while the solid line represents the real distance. It can be obtained that the system can obtain fairly good localization information of the drogue.

VI. Conclusions

In this research, a vision-based position estimation method for probe-and-drogue refueling systems is presented. These real-time detecting and matching algorithms are of strong robustness. Through the simulation, the effectiveness and accuracy of the proposed method have been demonstrated. Therefore, the proposed vision-based position

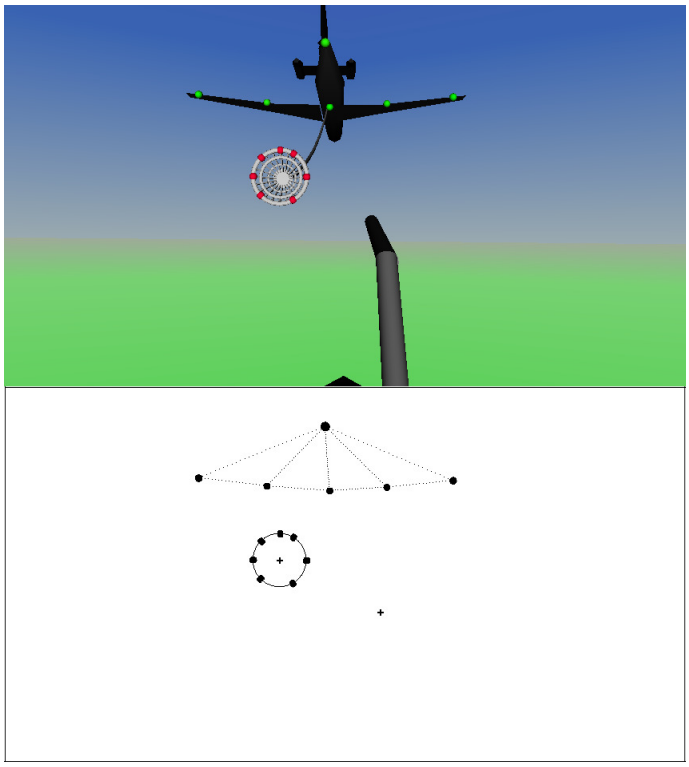

Fig. 7: All markers are identified properly

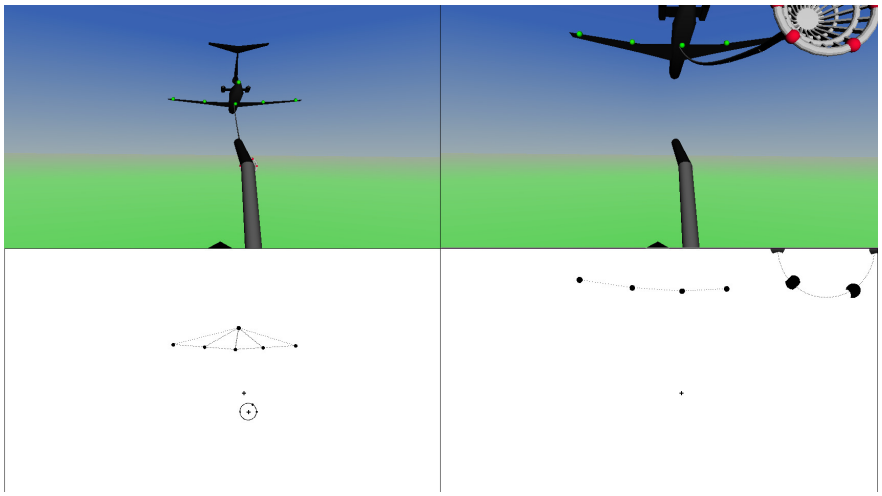

Fig. 8: 85% and 50% of the drogue is blocked.

estimation method is promising to acquire the relative distance information between the drogue and the receiver aircraft, which can be later applied to the autonomous aerial refueling system.

In future research, the proposed method could be tested with a real camera, such as in a hardware-in-the-loop simulation. Besides, some other extreme circumstances which may happen in real flight should be considered.

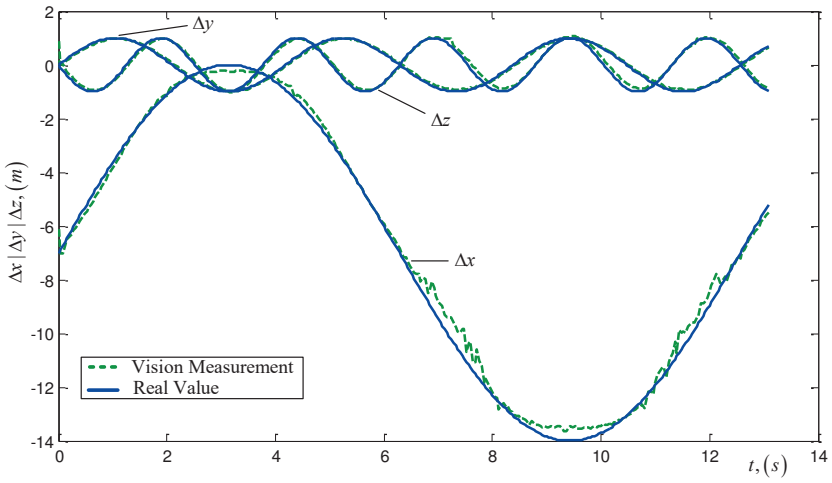

Fig. 9: The effect of visual location and tracking.

REFERENCES

[1] G. Campa, M. Mammarella, M. R. Napolitano, et al, "A comparison of pose estimation algorithms for machine vision based aerial refueling for UAVs", *Proceedings of the 2006 Conference on Control and Automation, Mediterranean*, 2006:1-6.

[2] S. M. Ross, M. D. Menza and E. T. Waddell, "Demonstration of a control algorithm for autonomous aerial refueling (Project "No Gyro")", *Technical report, Air Force Flight Test Center Edwards AFB CA*, 2005.

[3] J. L. Hansen, J. E. Murray and N. V. Campos, "The NASA Dryden AAR Project: A flight test approach to an aerial refueling system", *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, 2004, AIAA 2004-4939.

[4] W. R. Williamson, G. J. Glenn, V. T. Dang, et al, "Sensor fusion applied to autonomous aerial refueling", *Journal of Guidance, Control and Dynamics*, 2009, 32(1):262-275.

[5] J. N. Fu, Q. Fu, U. Arif, et al, "A pose estimation method of a moving target based on off-board monocular vision", *Guidance, Navigation and Control Conference (CGNCC), 2016 IEEE Chinese. IEEE*, 2017:2082-2087.

[6] J. Kimmett, J. Valasek and J. Junkins, "Autonomous aerial refueling utilizing a vision based navigation system", *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2002, AIAA 2002-4669.

[7] A. D. Weaver, "Using predictive rendering as a Vision-Aided technique for autonomous aerial refueling", *M.S. Thesis, Air Force Institute of Technology*, 2009.

[8] Q. Fu, Q. Quan and K. Y. Cai, "Robust pose estimation for multirotor UAVs using off-board monocular vision", *IEEE Transactions on Industrial Electronics*, 2017, 64(10): 7942-7951.

[9] S. Chen, H. B. Duan, Y. Deng, et al, "Drogue pose estimation for unmanned aerial vehicle autonomous aerial refueling system based on infrared vision sensor", *Optical Engineering*, 2017, 56(12): 124105.

[10] J. Valasek, K. Gunnam, J. Kimmett, et al, "Vision-based sensor and navigation system for autonomous air refueling", *Journal of Guidance, Control and Dynamics*, 2005, 28(5):979-989.

[11] D. Monish, Tandale, R. Bowers and J. Valasek, "Trajectory tracking controller for vision based probe and drogue autonomous aerial refueling", *Journal of Guidance, Control and Dynamics*, 2006, 29(4): 846-857.

[12] C. P. Lu, G. D. Hager and E. Mjolsness, "Fast and globally convergent pose estimation from video images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, 22(6): 610-622.

[13] Q. Quan, "Introduction to multicopter design and control", *Springer*, Singapore, 2017.

[14] Z. B. Wei, X. H. Dai, Q. Quan, et al, "Drogue dynamic model under bow wave effect in probe and drogue aerial refuelling", *IEEE Transactions on Aerospace and Electronic Systems*, 2016, 52(4): 1728-1742.

[15] P. Sturm, "Pinhole camera model", *Springer*, Boston, 2014.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[18] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[20] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "Dssd: Deconvolutional single shot detector," *arXiv preprint arXiv:1701.06659*, 2017.

[21] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[22] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[23] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "Cspnet: A new backbone that can enhance learning capability of cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 390–391.

[24] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8759–8768.

[25] Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, and W. Zuo, "Enhancing geometric factors in model learning and inference for object detection and instance segmentation," *IEEE transactions on cybernetics*, vol. 52, no. 8, pp. 8574–8586, 2021.

[26] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 12 993–13 000.