

# A Robust Real-time Embedded Vision System on an Unmanned Rotorcraft for Ground Target Following

Feng Lin, *Student Member, IEEE*, Xiangxu Dong, Ben M. Chen, *Fellow, IEEE*, Kai-Yew Lum, *Member, IEEE*, and Tong H. Lee *Member, IEEE*

**Abstract**—In this paper, we present a systematic design and implementation of a robust real-time embedded vision system for an unmanned rotorcraft for ground target following. The hardware construction of the vision system is presented, and the on-board software system is developed based on a multithread technique capable of coordinating multiple tasks. To realize the autonomous ground target following, a sophisticated feature-based vision algorithm is proposed by using an on-board color camera and navigation sensors. The vision feedback is integrated with the flight control system to guide the unmanned rotorcraft to follow a ground target in flight. The overall vision system has been tested in actual flight missions, and the results obtained show that the overall system is very robust and efficient.

**Index Terms**—real-time systems, vision systems, image processing, unmanned aerial vehicles, target detection and following.

## I. INTRODUCTION

UNMANNED aerial vehicles (UAVs) have recently aroused much interest in the civil and industrial markets, ranging from industrial surveillance, agriculture, academic research, to wildlife conservation [8], [14], [26], [6], [15], [32], [34]. Particularly, thanks to its vertical take-off-and-landing, hovering and maneuvering capabilities, the unmanned rotorcraft has received much attention in the defense and security community [1]. More specifically, an unmanned rotorcraft equipped with a vision payload can perform a wide range of tasks, such as search and rescue, surveillance, target detection and tracking, etc., as vision provides a natural sensing modality — in terms of human comprehension — for feature detection and tracking [28], [29]. Instead of vision being merely a payload, many research efforts have also been devoted to vision-aided flight control [2], [17], [22], tracking [25], [28], terrain mapping [27], and navigation [18], [23].

We note that most of the works reported in the literature, however, focus on only a certain part of vision systems for UAVs, such as hardware construction or vision algorithms. Many of them are adopted from those designed for ground robots, which are not very suitable for applications on UAVs. To the best of our knowledge, there is hardly any systematic documentation in the open literatures dealing with the

complete design and implementation of the vision system for unmanned rotorcrafts, which includes architectural and algorithmic design of real-time vision systems. In addition, although the target tracking in video sequences have already been studied in a number of applications, there has been very little research related to the implementation of vision-based target following for UAVs.

In this paper, we present the design and implementation of a comprehensive real-time embedded vision system for an unmanned rotorcraft, which includes an on-board embedded hardware system, a real-time software system and mission-based vision algorithms. More specifically, the on-board hardware system is designed to fulfill the image processing requirements by using the commercial off-the-shelf products, such as PC104 embedded modules. A real-time vision software is developed, which is running on the real-time operating system QNX. An advanced and efficient vision algorithm is then proposed and implemented to realize the ground target tracking, which is suited for the unmanned aerial vehicles. The proposed vision scheme is integrated with the on-board navigation sensors to estimate the relative distance between the target and the UAV. Finally, using the vision feedback, a two-layer target tracking control framework is utilized to control a pan/tilt servo mechanism to keep the target in the center of the image, and guide the UAV to follow the motion of the target.

The remainder of this paper is organized as follows: Sections II and III present respectively the development of hardware and software of the embedded vision system for a UAV, whereas coordinate systems adopted in the UAV vision systems are described in Section IV. Section V details the vision-based ground target detection and tracking algorithms, as well as the target following scheme based on vision signal feedback. The experimental results of the vision system obtained through actual flight tests are presented in Section VI. Finally, we draw some concluding remarks in Section VII.

## II. HARDWARE CONFIGURATION OF THE VISION SYSTEM

The hardware configuration of the proposed on-board vision system for the UAV, as illustrated in Figure 1, consists of the following five main parts: a visual sensor, an image acquisition module, a vision processing module, a pan/tilt servo mechanism, and a video- and data-link.

### A. Visual Sensor: Video Camera

A visual sensor is employed on-board to obtain in-flight visual information of the surrounding environment of the

Manuscript received November 9, 2010. Accepted for publication May 19, 2011.

Copyright © 2009 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

F. Lin, X. Dong, B. M. Chen and T. H. Lee are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576. Email: {linfeng, dong07, bmchen, eleleeth}@nus.edu.sg.

K. Y. Lum is with Temasek Laboratories, National University of Singapore, Singapore 119260. Email: tslumky@nus.edu.sg.

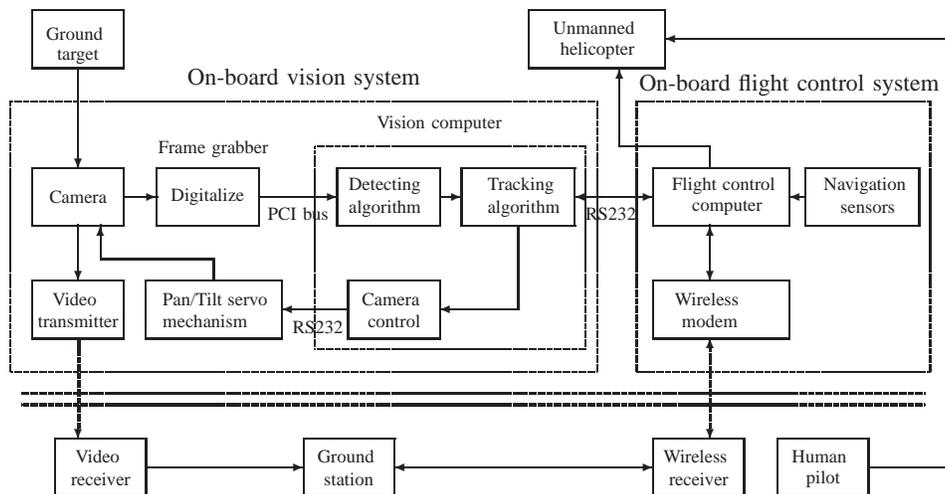


Fig. 1. The configuration of the overall vision system.

UAV. Interested visual information is composed of silent and dynamic features, such as the color and shape of land marks, and motions of vehicles. A color video camera is selected as the on-board visual sensor in our system, which has a compact size and a weight less than 30g, as well as 380 TV line resolution and 40-degree field of view.

#### B. Image Acquisition Module: Frame Grabber

The primary function of a frame grabber is to perform the A/D conversion of the analog video signals and then output the digitalized data to a host computer for further processing. Our selection is a PC/104(-plus)-standard frame grabber, a Colory 104, which has the following features: (1) high resolution: Colory 104 is capable of providing a resolution up to  $720 \times 576$  (pixels), which is sufficient for on-line processing; (2) multiple video inputs: it is able to collect data from multiple cameras; (3) sufficient processing rate: the highest A/D conversion rate is 30 frames per second (FPS), which is higher than the on-board vision processing rate (10 FPS); (4) featured processing method: two tasks are used alternatively to convert the digital video signal into specified formats.

#### C. Vision Processing Module: Vision Computer

As shown in Figure 1, the digitalized visual signals provided by the frame grabber is transferred to the on-board vision computer that is the key unit of the vision system. The vision computer coordinates the overall vision system, such as image processing, target tracking, and communicating with the flight control computer, which is to be described in detail later in Section V. In this work, the configuration of using two separated embedded computers in the on-board system for UAVs are proposed: one for flight control, and another one for machine vision algorithms. We choose such a configuration for on-board system because of the following reasons: (1) the computation consumption of flight control task and vision

program are very heavy, which can hardly be carried out together in a single embedded computer; (2) the sampling rate of the flight control computer is faster than the vision computer, since the faster sampling rate is required to stabilize the unmanned rotorcraft; (3) the decoupled structure reduces the negative effect of data blocking caused by the vision program and flight control system, and thus makes the overall system more reliable.

In the proposed vision system, a separated on-board PC104 embedded computer, Cool RoadRunner III, is employed to process the digitalized video signal and execute the vision algorithms. The core of the board is an Intel LV Pentium-III processor running at 933 MHz. A compact flash memory card is used to save the captured images.

#### D. Pan/Tilt Servo Mechanism

In the application of the ground target following, it is required to keep the target objects in the field of view of the camera to increase the flexibility of vision-based tracking. As such, we decide to mount the camera on a pan/tilt servo mechanism that can rotate in the horizontal and vertical directions.

#### E. Wireless Data Link and Video Link

In order to provide ground operators with clear visualization to monitor the work that the on-board vision is processing during flight tests, the video captured by the on-board camera is transmitted and displayed in a ground control station. An airborne 2.4 GHz wireless video link is used to transmit the live video captured to the ground control station.

### III. CONFIGURATION OF THE VISION SOFTWARE SYSTEM

Based on the proposed hardware system, the configuration of the on-board vision software system is presented. The

purpose of the vision software system is to coordinate the work of on-board devices and implement vision algorithms. Since the vision software system targets for real-time applications and runs in an embedded PC104 computer, QNX Neutrino, a real-time embedded operating system, is employed as the developing platform. QNX Neutrino has a microkernel that requires fewer system resources, and performs more reliably and efficiently for embedded systems during runtime compared to the traditional monolithic kernel.

The vision software program coordinates tasks such as capturing video, controlling pan/tilt servo mechanism, as well as performing the vision detecting and tracking algorithms. To make the vision software system easy to design and robust to perform, the entire vision software system is divided into several main blocks. Each block is assigned a special task.

- 1) CAM: Reading RGB24 image from the buffers assigned to the frame grabber. The reading rate is set up to be 10 FPS. In order to reduce the risk of damaging the image data, two buffers are used to store the captured images by the frame grabber alternatively.
- 2) IMG: Processing the captured images, carrying out the vision algorithms, such as the automatic tracking and camera control, which will be explained in Section V.
- 3) SVO: Controlling the rotation of the pan/tilt servo mechanism to keep the ground target in a certain location of the image.
- 4) SAV: Saving the captured and processed images to a high-speed compact flash.
- 5) COM: Communicating with the flight control computer. The flight control computer sends the states of the UAV and commands from the ground station to the vision computer, and the vision computer sends the estimated relative distance between the UAV and the ground target to the flight control computer to guide the flight of the UAV.
- 6) USER: Providing a mean for users to control the vision program such as running and stopping the tracking as well as changing the parameters of the vision algorithms.
- 7) MAIN: Managing and scheduling the work of the entire vision software system.

#### IV. COORDINATE FRAMES USED IN VISION SYSTEMS

Depicted in Figure 2 are coordinate systems adopted in the UAV vision systems. More specifically, we have

- 1) The local north-east-down (NED) coordinate system (labeled with a subscript 'n') is an orthogonal frame on the surface of the earth, whose origin is the launching point of the aircraft on the surface of the earth.
- 2) The body coordinate system (labeled with a subscript 'b') is aligned with the shape of the fuselage of the aircraft.
- 3) The servo-base coordinate system (labeled with a subscript 's') is attached to the base of the pan/tilt servo mechanism, which is aligned with the body coordinate system of the UAV.
- 4) The spherical coordinate system (labeled with a subscript 'sp') is also attached to the base of the pan/tilt

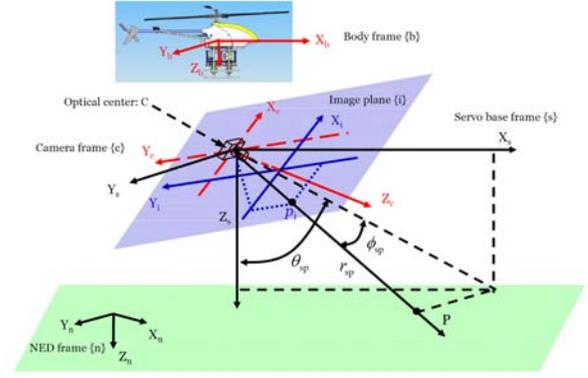


Fig. 2. Coordinate frames used in unmanned vision systems

servo mechanism. It is used to define the orientation of the camera and the target with respect to the UAV. Given a generic point  $\mathbf{p}_s = (x_s, y_s, z_s)^T$  in the servo-base coordinate system, its position can be defined in the spherical coordinate system by three numbers: radius  $r_{sp}$ , azimuth angle  $\theta_{sp}$  and elevation angle  $\phi_{sp}$ , which is given by

$$\mathbf{p}_{sp} = \begin{pmatrix} r_{sp} \\ \theta_{sp} \\ \phi_{sp} \end{pmatrix} = \begin{pmatrix} \sqrt{x_s^2 + y_s^2 + z_s^2} \\ \tan^{-1} \left( \frac{x_s}{z_s} \right) \\ \sin^{-1} \left( \frac{y_s}{r_{sp}} \right) \end{pmatrix} \quad (1)$$

- 5) The camera coordinate system (labeled with a subscript 'c'), whose origin is the optical center of the camera. The  $Z_c$ -axis is aligned with the optical axis of the camera and points from the optical center  $C$  towards the image plane.
- 6) The image frame (or the principle image coordinate system) (appended with a subscript 'i') has the origin at the principal point. The coordinate axes,  $X_i$  and  $Y_i$ , are aligned with the camera coordinate axes,  $X_c$  and  $Y_c$ , respectively.

#### V. VISION-BASED GROUND TARGET FOLLOWING

To realize the vision-based ground target detection, many vision approaches have been proposed worldwide, such as template matching [28], [7], background subtraction [19], [35], optical flow [18], [3], stereo vision based technologies [11], and feature-based approaches [31], [39], [20], [36].

In this paper, a sophisticated vision-based target detection and tracking scheme is proposed, as illustrated in Figure 3, which employs robust feature descriptors and efficient image tracking techniques. Based on the vision sensing data and navigation sensors, the relative distance to the target is estimated. Such estimation is integrated with the flight control system to guide the UAV to follow the ground target in-flight.

##### A. Target Detection

The purpose of the target detection is to identify the target of interest from the image automatically based on a database of

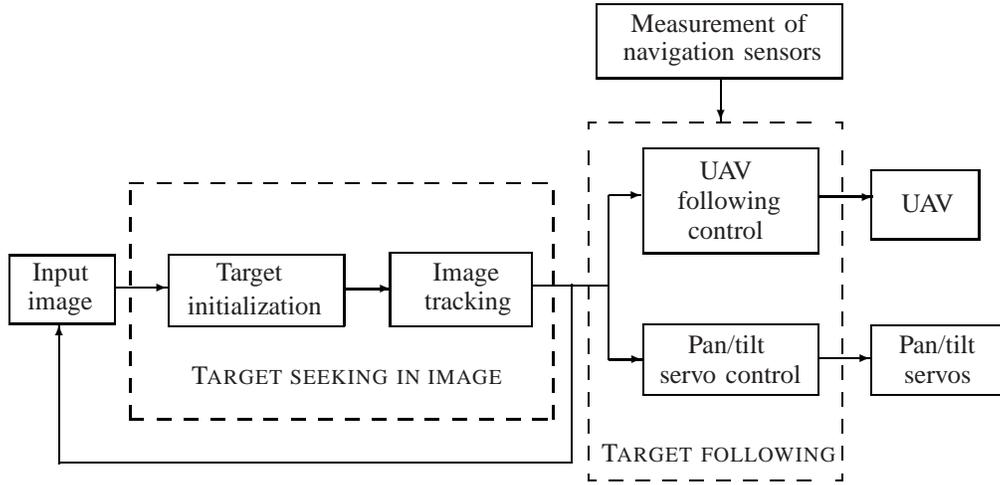


Fig. 3. Flow chart of the ground target detection, tracking and following

pre-selected targets. A toy car is chosen as the ground target. A classical pattern recognition procedure is used to identify the target automatically, which includes three main steps, i.e., segmentation, feature extraction, and pattern recognition.

1) *Segmentation*: The segmentation step aims to separate the objects of interest from background. To simplify the further processing, some assumptions are made. First, the target and environments exhibit *Lambertian reflectance*, and in other words, their brightness is unchanged regardless of viewing directions. Second, the target has a distinct color distribution compared to the surrounding environments.

*Step 1: Threshold in Color Space*. To make the surface color of the target constant and stable under the varying lighting condition, the color image is represented in HSV space, which stands for Hue (*hue*), Saturation (*sat*) and Value (*val*) introduced originally by Smith [33]. Pre-calculated threshold ranges are applied to the *hue*, *sat*, and *val* channels:

$$hue_r = [h_1, h_2], sat_r = [s_1, s_2], val_r = [v_1, v_2] \quad (2)$$

Only the pixel values falling in these color ranges are described as the foreground points, and pixels of the image that fall out of the specified color range are removed. The procedure of the image pre-process is illustrated in Figure 4.

*Step 2: Morphological Operation*. As shown in Figure 4, normally, the segmented image is not smooth and has many noise points. Morphological operations are then employed to filter out noise, fuses narrow breaks and gulfs, eliminates small holes, and fill gaps in the contours. Next, a contour detection approach is used to obtain the complete boundary of the objects in the image, which will be used in the feature extraction.

2) *Feature Extraction*: Generally, multiple objects will be found in the segmented images, including the true target and false objects. The geometric and color features are used as the descriptors to identify the true target.

*Geometry Feature Extraction*: To describe the geometric features of the objects, the four lowest moment invariants proposed in [25] are employed, since they are independent of position, size and orientation in the visual field. The four

lowest moment invariants, defined in the segmented image  $I(x, y)$ , are given by

$$\phi_1 = \eta_{20}^m + \eta_{02}^m \quad (3)$$

$$\phi_2 = (\eta_{20}^m - \eta_{02}^m)^2 + 4(\eta_{11}^m)^2 \quad (4)$$

$$\phi_3 = (\eta_{30}^m - 3\eta_{12}^m)^2 + (\eta_{03}^m - 3\eta_{21}^m)^2 \quad (5)$$

$$\phi_4 = (\eta_{30}^m + \eta_{12}^m)^2 + (\eta_{03}^m + \eta_{21}^m)^2 \quad (6)$$

where  $\eta_{pq}^m$ , for  $p + q = 2, 3, \dots$ , is the improved normalized central moment defined as

$$\eta_{pq}^m = \frac{\mu_{pq}^c}{A^{(p+q+1)/2}} \quad (7)$$

where  $A$  is the interior area of the shape, and  $\mu_{pq}^c$  is the central moment defined as

$$\mu_{pq}^c = \int_C (x - \bar{x})^p (y - \bar{y})^q ds, \quad p, q = 0, 1, \dots \quad (8)$$

Note that in (8),  $C$  is the boundary curve of the shape,  $\int_C$  is a line integral along  $C$ ,  $ds = \sqrt{(dx)^2 + (dy)^2}$ , and  $[\bar{x}, \bar{y}]$  is the coordinate of the centroid of the shape in the image plane.

In addition, compactness is another useful feature descriptor for recognition. Compactness of a shape is measured by the ratio of the square root of the area and the perimeter, which is given by

$$\text{Compactness: } \beta_c = \frac{\sqrt{A}}{C}. \quad (9)$$

It can be easily proven that compactness is invariant with respect to translation, scaling and rotation.

*Color Feature Extraction*: To make the target detection and tracking more robust, we also employ color histogram to represent the color distribution of image area of the target, which is not only independent of the target orientation, position and size, but also robust to partial occlusion of the target and easy to implement. Due to the stability in outdoor environments, only *hue* and *val* are employed to construct the color histogram for object recognition, which is defined as:

$$H = \{hist(i, j)\}, \quad i = 1, \dots, N_{hue}, \quad j = 1, \dots, N_{val} \quad (10)$$

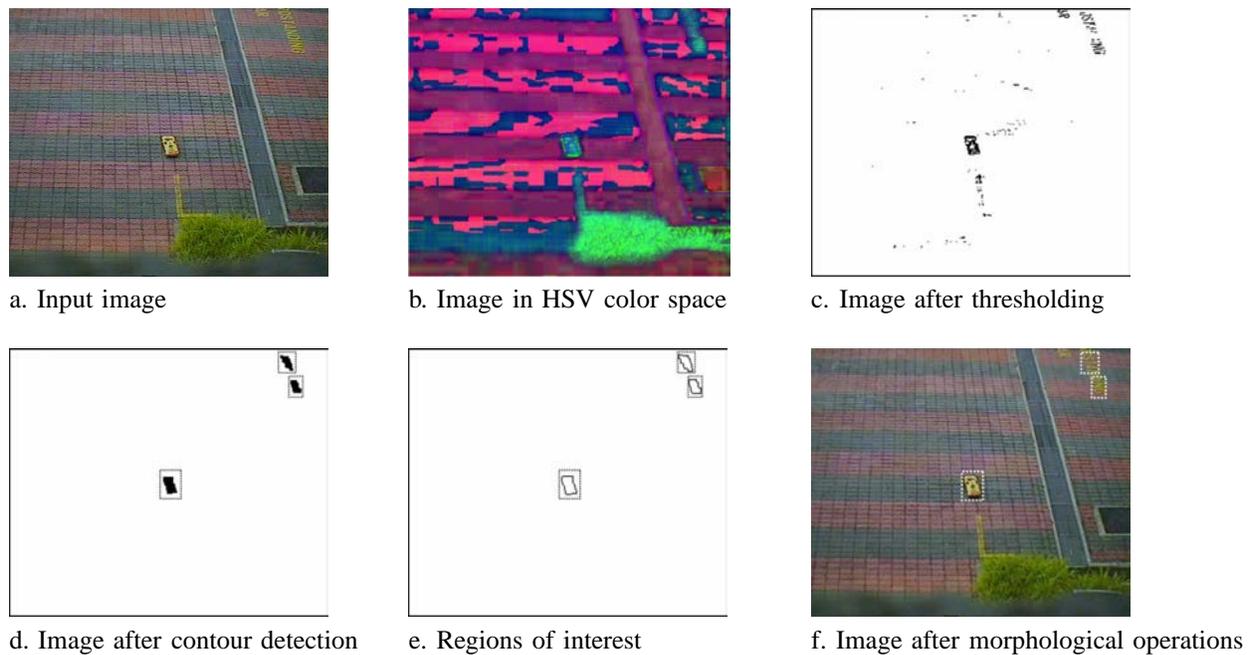


Fig. 4. Illustration of Segmentation.

where

$$hist(i, j) = \sum_{(x, y) \in \Omega} \delta \left( i, \left[ \frac{hue(x, y)}{N_{hue}} \right] \right) \delta \left( j, \left[ \frac{val(x, y)}{N_{val}} \right] \right),$$

and where  $N_{hue}$  and  $N_{val}$  are the partition numbers of  $hue$  and  $val$  channels, respectively,  $\Omega$  is the region of the target,  $[\cdot]$  is the nearest integer operator, and  $\delta(a, b)$  is the Kronecker delta function.

**Dynamic Features:** Besides the static features extracted from the foreground objects, we further calculate their dynamic motion using the Kalman filtering technique. The distance between the location of each object  $z_i$  and the predicted location of the target  $\hat{z}$  is employed as a dynamic feature. The detailed procedure for predicting the location of the target in the image is to be discussed in Section V-B1. Both the static and dynamic features of them are then employed in the pattern recognition.

The extracted features of an object need to be arranged in a compact and identifiable form [30]. A straightforward way is to convert these features in a high dimensional vector. For example, the feature vector of  $i$ -th object is given by

$$\begin{aligned} \alpha_i &= [\beta_{c,i}, \phi_{1,i}, \phi_{2,i}, \phi_{3,i}, \phi_{4,i}, H_i, \mathbf{z}_i] \\ &= \{\alpha_{k,i}\}, \quad k = 1, \dots, d \end{aligned} \quad (11)$$

where  $d$  is the dimension of the feature vector.

**3) Pattern Recognition:** The purpose of the pattern recognition is to identify the target from the extracted foreground objects in terms of the extracted features in (11). The straightforward classifier is to use the nearest-neighbor rule. It calculates a metric or ‘distance’ between an object and a template in a feature space, and assign the object to the class with the highest score. But to take advantage of *a priori* knowledge of the feature distribution, the classification problem is formulated under the model-based framework,

and solved by using a probabilistic classifier. A discriminant function, derived from Bayes’ theorem, is employed to identify the target. This function is computed based on the measured feature values of each object and the known distribution of features obtained from training data.

**Step 1. Pre-filter:** Before classifying the objects, a pre-filter is carried out to remove the objects whose feature values are outside certain regions determined by *a priori* knowledge. This step aims to improve the robustness of the pattern recognition and speed up the calculation.

**Step 2. Discriminant Function:** We use the discriminant function, derived from Bayes’ theorem, to determine the target based on the measured feature values of each object and the known distribution of features of the target obtained from training data. We assume these features are independent and fulfill normal distributions. Thus, we can define the simplified discriminant function with weightings as

$$f'_j(\alpha_i) = \sum_{k=1}^5 w_k \left( \frac{\alpha_{k,i} - \mu_{k,j}}{\sigma_{k,j}} \right)^2 + w_6 \left( \frac{d_c(H_i, G_j) - \mu_{6,j}}{\sigma_{6,j}} \right)^2$$

where

$$d_c(H_i, G_j) = \frac{\sum_{p=1}^{N_h} \sum_{q=1}^{N_v} \min(H_i(p, q), G_j(p, q))}{\min(|H_i|, |G_j|)}, \quad (12)$$

and  $\alpha_{k,i}$  is the  $k$ -th element of the feature vector of the object  $i$ .  $\mu_{k,j}$  and  $\sigma_{k,j}$  are the mean and standard deviation of the distribution of the corresponding feature.  $G_j$  is the color histogram template of a pre-defined target. In fact, the location information is not used in the detection mode. The target  $i$  with the minimum value is considered as the candidate target.  $w_1$  to  $w_6$  are the weighting scalars of the corresponding features. In terms of the likelihood values of the objects, a decision rule

is defined as:

$$D = \begin{cases} \text{target} = \arg \min_i f'_j(\alpha_i), \text{ which belongs to class } j, & \text{if } \min f'_j(\alpha_i) \leq \Gamma'_j \\ \text{no target in the image,} & \text{if } \min f'_j(\alpha_i) > \Gamma'_j \end{cases}$$

where  $\Gamma'$  is a threshold value chosen based on training data. This decision rule chooses the object, say  $i$ , with the smallest value of the simplified discriminant function as the candidate target. If  $f'_j(\alpha_i) < \Gamma'_j$ , then the scheme decides that object  $i$  is the target. Otherwise, the scheme indicates that there is no target in the current image.

### B. Image Tracking

As shown in Figure 3, after initialization, the image tracking techniques are employed. The purpose of image tracking is to find the corresponding region or point to the given target. Unlike the detection, the entire image search is not required. Thus, the processing speed of image tracking is faster than the detection. The image tracking problem can be solved by using two main approaches, 1) filtering and data association, and 2) target representation and localization [13].

*Filtering and Data Association:* The filtering and data association approach can be considered as a top-down process. The purpose of the filtering is to estimate the states of the target, such as static appearance and location. Typically, the state estimation is achieved by using filtering technologies [38], [40]. It is known (see, for example, [24]) that most of tracking algorithms are model based because a good model-based tracking algorithm will greatly outperform any model-free tracking algorithm if the underlying model is found to be a good one. If the measurement noise satisfied the Gaussian distribution, the optimal solution can be achieved by the Kalman filtering technique [4]. In some more general cases, particle filters are more suitable and robust [21]. However, the computational cost increases and the sample degeneracy is also a problem. When multiple targets are tracked in the image sequence, the validation and association of the measurements become a critical issue. The association techniques, such as Probabilistic Data Association Filter (PDAF) and Joint Probabilistic Data Association Filter (JPDAF) are widely used [37].

*Target Representation and Localization:* Besides using the motion prediction to find the corresponding region or point, Target Representation and Localization is considered as another efficient way, which is referred to as a bottom-up approach. Among the searching methods, the mean shift approach using the density gradient is commonly used [5], which is trying to search the peak value of the object probability density. However, the efficiency will be limited when the spatial movement of the target become significant.

To take advantages of the aforementioned approaches, using multiple trackers are widely adopted in applications of image tracking. In [37], the tracking scheme by integrating motion, color and geometric features were proposed to realize robust image tracking. In conclusion, combining the motion filtering and advanced searching algorithms will definitely make the tracking processing more robust, but the computational load is heavier.

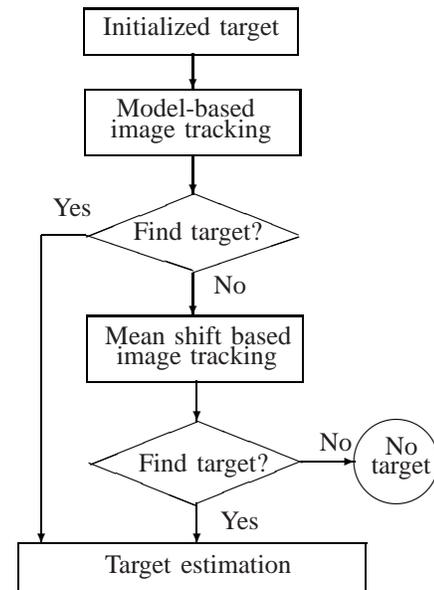


Fig. 5. Flow chart of image tracking

In our approach, instead of using multiple trackers simultaneously, a hierarchical tracking scheme is proposed to balance the computational cost and performance, which is illustrated in Figure 5. In the model-based image tracking, the Kalman filtering technique is employed to provide accurate estimation and prediction of the position and velocity of a single target, referred to as dynamic information. If the model-based tracker fails to find the target, a mean shift based image tracking method will be activated to retrieve the target back in the image.

1) *Model-based Image Tracking:* Model-based image tracking will predict the possible location of the target in the subsequent frames, and then do the data association based on an updated likelihood function. The advantage of the model-based image tracking is to combine dynamic features with geometric features of the target in the image tracking under noise and occlusion condition. In addition, several methods are employed to make the tracking more robust and efficient, which are given by:

- 1) Narrow the search window in terms of the prediction of the Kalman filter;
- 2) Integrate the spatial information with appearance and set the different weightings for the discriminant function.

The motion of the centroid of the target,  $\mathbf{x} = [\bar{x}, \dot{\bar{x}}, \bar{y}, \dot{\bar{y}}]^T$ , in the two-dimensional image coordinate is tracked using a 4th-order Kalman filter, which predicts the possible location of the target in the successive frames. The discrete-time model of the target motion can be expressed as

$$\begin{aligned} \mathbf{x}(k|k-1) &= \Phi \mathbf{x}(k-1) + \Lambda \mathbf{w}(k-1), \\ \mathbf{z}(k) &= \mathbf{H} \mathbf{x}(k) + \mathbf{v}(k), \end{aligned} \quad (13)$$

where  $\mathbf{w}$  and  $\mathbf{v}$  denote the input and measurement zero-mean Gaussian noises

$$\Phi = \begin{bmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix}, \Lambda = \begin{bmatrix} \frac{T_s^2}{2} & 0 \\ T_s & 0 \\ 0 & \frac{T_s^2}{2} \\ 0 & T_s \end{bmatrix},$$

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$T_s$  is the sampling period of the vision-based tracking system. A Kalman filter can then be designed based on the above motion model to estimate the states of the target in the image plane. The filter consists of the following stages:

1) Predicted state

$$\hat{\mathbf{x}}(k|k-1) = \Phi \hat{\mathbf{x}}(k-1)$$

2) Updated state estimate

$$\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)(\mathbf{z}(k) - \mathbf{H}\hat{\mathbf{x}}(k|k-1))$$

where  $\mathbf{K}(k)$  is the optimal Kalman gain.

The distance between the location of each object  $\mathbf{z}_i$  and the predicted location of the target  $\hat{\mathbf{z}}$  is employed as the dynamic feature defined by

$$\tilde{\mathbf{z}}_i = \mathbf{z}_i(k) - \hat{\mathbf{z}}(k) = \mathbf{z}_i(k) - \mathbf{H}\hat{\mathbf{x}}(k|k-1)$$

Thus, the updated discriminant function, which includes the appearance and spatial information, is shown as follows:

$$f'_j(\alpha_i) = \sum_{k=1}^5 w_k \left( \frac{\alpha_i(k) - \mu_j(k)}{\sigma_j(k)} \right)^2 + w_6 \left( \frac{d_c(H_i, G_j) - \mu_j(6)}{\sigma_j(6)} \right)^2 + w_7 \left( \frac{\|\tilde{\mathbf{z}}_i\| - \mu_j(7)}{\sigma_j(7)} \right)^2 \quad (14)$$

Most of time, the model-based tracker can lock the target in the image sequence, but sometime it may fail due to the noise or disturbance, such as partial occlusion. Thus, a scheme is required to check whether the target is still in the image, and then activate other trackers.

2) *Switching Mechanism*: The purpose of the switching mechanism is to check whether the target is still in the image when the target is lost by the model-based tracker. If yes, the mean shift tracker will be activated. The lost of the target can be attributed to the poor match of features due to noise, distortion, or occlusion in the image. An alternative reason may be the maneuvering motion of the target, and the target is out of the image. Therefore, in order to know the reason and take the special way to find target again, it is necessary to formulate the decision making as the following hypothesis testing problem:

$H_0$  : The target is still in the image;

$H_1$  : The target is not in the image due to maneuvers.

The estimation error is considered as a random variable, which is defined by:

$$\varepsilon = (\mathbf{H}\hat{\mathbf{x}}_{k-1} - \mathbf{z}_{k-1})' \Sigma^{-1} (\mathbf{H}\hat{\mathbf{x}}_{k-1} - \mathbf{z}_{k-1})$$

where  $\mathbf{H}\hat{\mathbf{x}}_{k-1} - \mathbf{z}_{k-1}$  is assumed to be  $N(0, \Sigma)$ -distributed.  $\varepsilon$  is Chi-square distributed with 2-degrees of freedom ( $x$  and  $y$  directions) under  $H_0$ .

$$\begin{cases} \varepsilon < \lambda = \chi_2^2(\alpha), & \text{if } H_0 \text{ is true} \\ \varepsilon \geq \lambda = \chi_2^2(\alpha), & \text{if } H_1 \text{ is true} \end{cases}$$

where  $1 - \alpha$  is the level of confidence, which should be sufficient high (for our system,  $1 - \alpha = 99\%$ ). If  $H_0$  is true, the Chi-square testing-based switching declares the target is still in the image and enables the mean shift based tracker.

3) *Mean Shift Based Image Tracking*: If the target is still in the image, Continuously Adaptive Mean Shift (CAMSHIFT) algorithm [5] is employed, which is shown in Figure 5. This algorithm uses the mean shift searching method to efficiently obtain the optimal location of the target in the search window. The principle idea is to search the dominated peak in the feature space based on the previous information and certain assumptions. The detected target is verified by comparing with an adaptive target template. The CAMSHIFT algorithm consists of three main steps: back projection, mean shift searching, and search window adaptation.

*Step 1. Back Projection*: In order to search the target in the image, the probability distribution image needs to be constructed based on the color distribution of the target. The color distribution of the target defined in *hue* channel is given by

$$hist_{tg}(i) = \sum_{(x, y) \in \Omega} \delta \left( i, \left[ \frac{hue_{tg}(x, y)}{N_{hue}} \right] \right), \quad i = 1, \dots, N_{hue}.$$

Based on the color model of the target, the back projection algorithm is employed to convert the color image to the color probability distribution image. The probability of each pixel  $\mathbf{I}_p(x, y)$  in the region of interest  $\Omega_r$  is calculated based on the model of the target, which is used to map the histogram results and given by

$$\mathbf{I}_p(x, y) = hist_{tg} \left( \left[ \frac{\mathbf{I}_{hue}(x, y)}{N_{hue}} \right] \right), \quad (15)$$

where  $\mathbf{I}_{hue}$  is the pixel values of the image in the *hue* channel.

*Step 2. Mean Shift Algorithm*: Based on the obtained color density image, a robust non-parametric method, the mean-shift algorithm, is used to search the dominated peak in the feature space. The mean-shift algorithm is an elegant way of identifying these locations without estimating the underlying probability density function [12].

Recall the discrete 2D image probability distributions in (15), the mean location (the centroid) of the search window is computed by

$$x_c(k) = \frac{\mathbf{M}_{10}}{\mathbf{M}_{00}}, \quad y_c(k) = \frac{\mathbf{M}_{01}}{\mathbf{M}_{00}},$$

where  $k$  is the number of iterations,

$$\mathbf{M}_{00} = \sum_{(x, y) \in \Omega_w} \mathbf{I}_p(x, y),$$

$$\mathbf{M}_{10} = \sum_{(x, y) \in \Omega_w} \mathbf{I}_p(x, y)x, \quad \mathbf{M}_{01} = \sum_{(x, y) \in \Omega_w} \mathbf{I}_p(x, y)y$$

and  $\Omega_w$  is the region of the search window;  $\mathbf{M}_{00}$  is the zeroth moment;  $\mathbf{M}_{10}$  and  $\mathbf{M}_{01}$  are the first moments for  $x$  and  $y$ , respectively. The search window is centered at the mean location  $\mathbf{c}(k) = (x_c(k), y_c(k))$ . Step 2 is to be repeated until  $\|\mathbf{c}(k) - \mathbf{c}(k-1)\| < \varepsilon$ .

*Step 3. Search Window Adaptation:* The region of interest is calculated dynamically using the motion filtering given in Section V-B1. To improve the performance of the CAMSHIFT algorithm, multiple search windows in the region of interest are employed. The initial locations and sizes of the searching windows are adopted from the centers and boundaries of the foreground objects respectively. These foreground objects are obtained using the color segmentation in the region of interest. In the CAMSHIFT algorithm, the size of the search window will be dynamically updated according to the moments of the region inside the search window [5]. Generally, more than one target candidate will be detected due to multiple search windows adopted. To identify the true target, the similarity between the target model and the detected target candidate is measured using the intersection comparison (12). This verification can effectively reduce the risk of detecting the false target.

### C. Target Following Control

We proceed to design a comprehensive target following system in this section. It consists of two main layers, the pan/tilt servo mechanism control and the UAV following control. The overall structure of the target following control is depicted in Figure 6. As mentioned in Section II, a pan/tilt servo mechanism is employed in the first layer to control the orientation of the camera to keep the target in an optimal location in the image plane, namely eye-in-hand visual serving [9], [10], which makes target tracking in the video sequence more robust and efficient. The parameters associated with the pan/tilt servo control in Figure 6 are to be introduced in detail later. In the second layer, the UAV is controlled to maintain a constant relative distance between the moving target and the UAV in-flight.

*1) Control of the Pan/Tilt Servo Mechanism:* As depicted in Figure 6, given a generic point  $\mathbf{P}$ ,  $\mathbf{p}_i$  and  $\mathbf{p}_i^*$  are the measured and desired locations of the projected point  $\mathbf{P}$  in the image plane, respectively.  $\mathbf{e} = [e_\phi, e_\theta]^T$  is the tracking error,  $\mathbf{u} = [u_\phi, u_\theta]^T$  is the output of the tracking controller,  $\mathbf{v} = [v_\phi, v_\theta]^T$  is the output of the pan/tilt servo mechanism.  $M$  is the camera model, which maps the points in the 3D space to the projected points in the 2D image frame.  $N$  is a function to calculate the orientation of an image point  $\mathbf{p}_i$  with respect to the UAV under the current  $\mathbf{v}$ . As mentioned in the definitions of the coordinate systems, the orientation of  $\mathbf{P}$  with respect to the UAV can be defined using azimuth and elevation angle in the spherical coordinate system, which is described by two rotation angles  $\mathbf{p}_e = [p_\phi, p_\theta]^T$ .

In image processing, the distortion of the lens is compensated, and the origin of the image plane is set as the principle point. Thus, we can obtain a simplified pinhole projection

model as

$$\begin{pmatrix} \mathbf{P}_i \\ 1 \end{pmatrix} = \frac{1}{\lambda} \begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{p}_c \quad (16)$$

with

$$\mathbf{p}_c = \mathbf{R}_{c/n} \mathbf{p}_n + \mathbf{t}_{c/n}, \quad (17)$$

where  $\lambda = z_c$  is the depth of the point  $\mathbf{P}$  in the camera coordinate system; and  $f_x$  and  $f_y$  are respectively the vertical and horizontal focal lengths in pixels;  $\mathbf{R}_{c/n}$  and  $\mathbf{t}_{c/n}$  are respectively the rotation matrix and the translation vector, which define the rigid-body transformation from the NED frame to the camera frame. Thus, we can define  $M$  as

$$\mathbf{p}_i = M(\mathbf{p}_n, \mathbf{v}) = \frac{1}{\lambda} \begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \end{bmatrix} \mathbf{R}_{c/n} (\mathbf{p}_n - \mathbf{t}_{c/n}).$$

Next, to derive the function  $N$ , we write the transformation between the camera coordinate system and the servo-base coordinate system as

$$\mathbf{p}_s = \mathbf{R}_{s/c}(\mathbf{v}) \mathbf{p}_c, \quad (18)$$

where  $\mathbf{p}_s$  is the coordinate of the point  $\mathbf{P}$  relative to the servo-base coordinate system;  $\mathbf{R}_{s/c}$  describes the rotation from the servo-base frame to the camera frame. We can then combine (18) with (16), and define the coordinate of the target in the spherical coordinate system

$$\mathbf{p}_e = \begin{pmatrix} p_\phi \\ p_\theta \end{pmatrix} = N(\mathbf{p}_i, \mathbf{v}) = \begin{pmatrix} \sin^{-1} \left( \frac{\bar{y}_s}{\bar{r}_{sp}} \right) \\ \tan^{-1} \left( \frac{\bar{x}_s}{\bar{z}_s} \right) \end{pmatrix}, \quad (19)$$

where

$$\begin{pmatrix} \bar{x}_s \\ \bar{y}_s \\ \bar{z}_s \end{pmatrix} = \mathbf{R}_{s/c}(\mathbf{v}) \begin{bmatrix} f_x^{-1} & 0 & 0 \\ 0 & f_y^{-1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} \\ \bar{r}_{sp} = \sqrt{\bar{x}_s^2 + \bar{y}_s^2 + \bar{z}_s^2} \quad (20)$$

The pan/tilt servo mechanism can be approximately considered as two decoupled servo motors, which regulate the visual sensor for horizontal and vertical rotation, respectively. The dynamic model of the servo motor can be described by using a standard second order system. Before proceeding to design the control law for the pan/tilt servo mechanism, we define the tracking error function as

$$\mathbf{e}(k) = \mathbf{p}_e - \mathbf{p}_e^* = N(\mathbf{p}_i(k), \mathbf{v}(k)) - N(\mathbf{p}_i^*, \mathbf{v}(k)), \quad (21)$$

where  $\mathbf{p}_e^*$  denotes the desired orientation of the camera. The control inputs will be sent to the pan/tilt servos after the vision-based target detection algorithm, which generally cost about one sampling period. To track the moving target efficiently, we calculate the pan/tilt servo control inputs using the predicted location of the target in the subsequent frame, which is derived from (13) and given by

$$\hat{\mathbf{p}}_i(k+1) = \hat{\mathbf{z}}(k+1|k) = \mathbf{H}\hat{\mathbf{x}}(k+1|k) \quad (22)$$

In implementation, it is not easy to measure the output of the pan/tilt servo  $\mathbf{v}$  in (21). We assume that the bandwidth of the

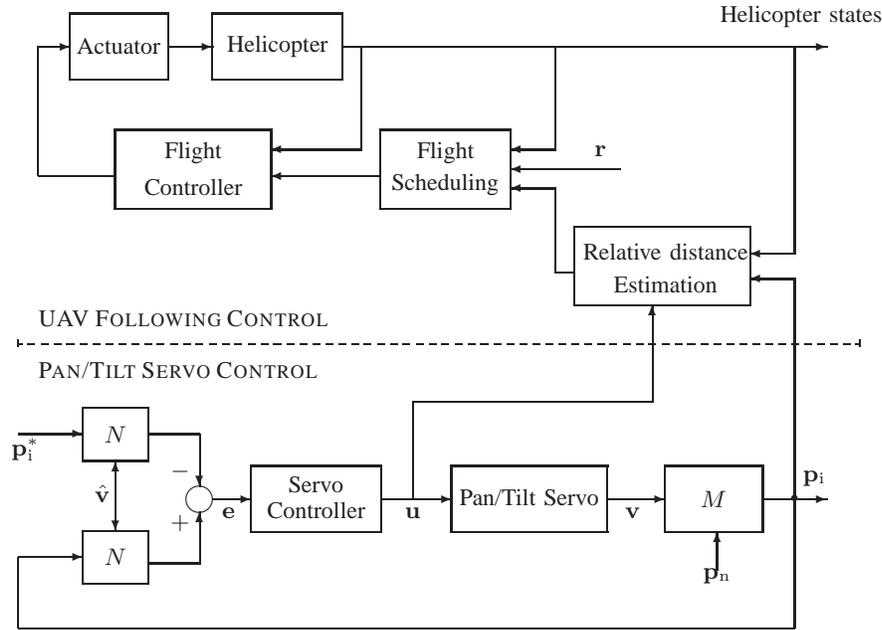


Fig. 6. Block diagram of the tracking control scheme.

pan/tilt servo mechanism is much faster than that of the control system. We then can ignore the transient of the pan/tilt servos, and consider them as scaling factors with one step delay. The estimate of  $\mathbf{v}$  is defined as

$$\hat{\mathbf{v}}(k) = K_d \mathbf{u}(k-1) \quad (23)$$

Replacing  $\mathbf{v}$  and  $\mathbf{p}_i$  with  $\hat{\mathbf{v}}$  and  $\hat{\mathbf{p}}_i$  in (21), we then can obtain the modified error function as

$$\mathbf{e}(k) = N(\hat{\mathbf{p}}_i(k+1), \hat{\mathbf{v}}(k)) - N(\mathbf{p}_i^*, \hat{\mathbf{v}}(k)) \quad (24)$$

The purpose of the design of the tracking control law is to minimize the tracking error function given in (24) by choosing a suitable control input  $\mathbf{u}(k)$ . Since the dynamics model of the pan/tilt servos is relatively simple, we employ a discrete-time proportional-integral (PI) controller (see, for example, [16]), which is structurally simple but fairly robust. It is very suitable for our real-time application. The incremental implementation of the PI controller is given by

$$\Delta \mathbf{u}(k) = K_p [\mathbf{e}(k) - \mathbf{e}(k-1)] + \frac{K_p T_s}{T_i} \mathbf{e}(k),$$

where the proportional gain and the integral time are chosen as  $K_p = 0.65$  and  $T_i = 0.8$ , respectively. We note that two identical controllers are respectively used for the pan and tilt servos, since the dynamics of the two servos are very close.

## 2) Following Control of the Unmanned Aerial Vehicle:

As illustrated in Figure 6, to estimate the relative distance between the target and the UAV, we combine the camera model (16) with the transformation in (17), and generate the overall geometric model from an ideal image to the NED frame:

$$\mathbf{p}_n = \lambda \mathbf{R}_{n/c} \begin{bmatrix} f_x^{-1} & 0 & 0 \\ 0 & f_y^{-1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} + \mathbf{t}_{n/c} \quad (25)$$

We assume that the ground is flat, and the height of the UAV to the ground:  $h$  is known. We have

$$\mathbf{R}_{n/c} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}, \quad \mathbf{t}_{n/c} = \begin{pmatrix} x_{n/c} \\ y_{n/c} \\ z_{n/c} \end{pmatrix} \quad (26)$$

which can be calculated by using the measurements of the on-board navigation sensors. Based on the assumption that the target is on the ground,  $z_n$  is equal to zero. We then can derive  $\lambda$  as

$$\lambda = \frac{-z_{n/c}}{r_7 \frac{x_i}{f_x} + r_8 \frac{y_i}{f_y} + r_9}$$

which together with (25) yields

$$\begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} = \begin{pmatrix} \lambda(r_1 \frac{x_i}{f_x} + r_2 \frac{y_i}{f_y} + r_3 + x_{n/c}) \\ \lambda(r_4 \frac{x_i}{f_x} + r_5 \frac{y_i}{f_y} + r_6 + y_{n/c}) \\ 0 \end{pmatrix} \quad (27)$$

As shown in Figure 6, the relative distance between the target and the UAV is estimated, which is employed as the reference signal to guide the UAV to follow the motion of the target. The tracking reference for the UAV is defined as

$$\begin{pmatrix} x_{uav} \\ y_{uav} \\ z_{uav} \\ \psi_{uav} \end{pmatrix}_{\text{ref}} = \begin{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{R}_{n/b} \begin{pmatrix} c_x \\ c_y \\ 0 \end{pmatrix} \\ h_0 \\ \psi_0 \end{pmatrix}$$

where  $c_x$  and  $c_y$  are the desired relative distance between the target and the UAV in the  $X_b$ - and  $Y_b$ -axis respectively;  $h_0$  is the pre-defined height of the UAV above the ground;  $\psi_0$  is

TABLE I  
EXPERIMENT RESULTS OF TARGET DETECTION AND TRACKING IN FLIGHT

Test No.	Total time (s)	Total frames	Target frames detected	Accuracy
1	101.8	761	728	95.66%
2	77.2	591	518	87.65%
3	50.4	388	382	98.45%
4	75.3	572	501	87.59%
5	86.4	662	645	97.43%

the predefined heading angle of the UAV;  $\mathbf{R}_{n/b}$  is the rotation matrix from the body frame to the local NED frame, which can be calculated in terms of the output of the on-board navigation sensors.

## VI. EXPERIMENTAL RESULTS

To verify the proposed vision system, multiple tests of the complete system were conducted. During these tests, the proposed vision-based unmanned helicopter: SheLion was hovering autonomously at a certain position. If the moving target entered into the view of the on-board camera, the target would be identified and tracked in the video sequence by the vision system automatically. Based on the vision information, the pan/tilt servo mechanism was controlled to keep the target in a certain position in the image as described in Section V-C1. The operator, then, can command the UAV to enter into the following mode, in which the UAV followed the motion of the target autonomously based on the estimated relative distance, using the algorithm proposed in Section V-C2.

The experimental results of the vision-based target detection and tracking in-flight are shown in Table I, which indicate that the proposed vision algorithm could effectively identify and track the target in the video sequence in the presence of the disturbance of unknown motion between the UAV and the target. One example of the pan/tilt servo tracking control in-flight is also shown in Figure 7. The solid line in Figure 7 indicates the expected position of the target in the image, and the dash line indicates the actual location of the target in image during the flight test. From Figure 7, we can observe that in spite of the unknown motion between the UAV and the target, the pan/tilt servo mechanism can effectively control target in a box-like neighborhood of the center point of the image by employing the vision-based pan/tilt servo control.

In the flight tests, the relative distance between the target and the UAV was estimated using the approach presented earlier, which is shown in Figure 8. The relative distance is also measured using the GPS receiver. The experimental results in Figure 8 indicate that the vision sensor can provide acceptable relative distance estimates between the UAV and the target based on the altitude information of the UAV and the location of the target in the image.

One example of the ground target following is described in Figure 9. In the experiment, the target was manually controlled to move randomly on the flat ground and the UAV followed the motion of the target automatically based on the scheme proposed in the previous sections. From Figure 9, we observe that the UAV can follow the trajectory of the target and keep the constant relative distance between the UAV and the target.

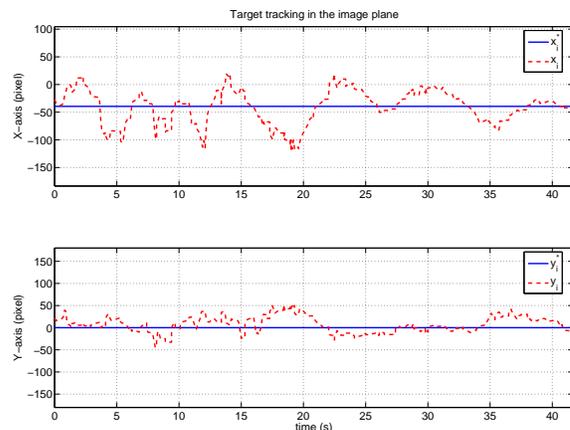


Fig. 7. The test result of the vision-based servo following

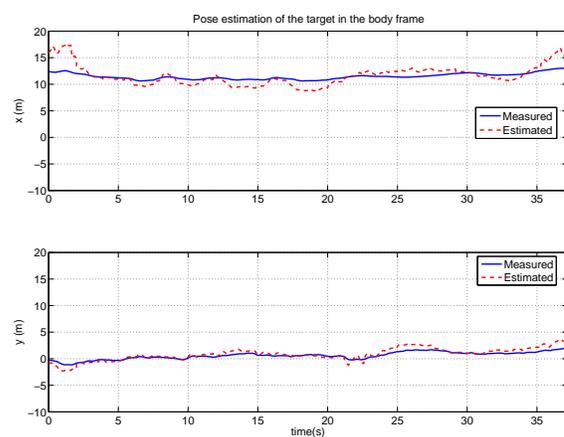


Fig. 8. The test result of the relative distance estimation

The results for the moving ground target following of the UAV indicate the efficiency and robustness of the proposed vision-based following scheme. The videos of the vision-based target following test is available upon requested.

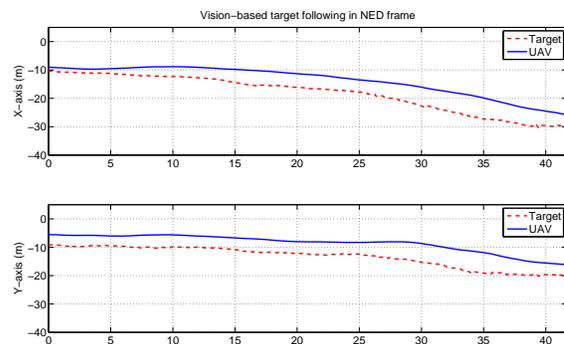


Fig. 9. The test result of the vision-based target following

## VII. CONCLUSION

In this paper, we have presented the comprehensive design and implementation of the vision system for the UAV, including hardware construction, software development, and an advanced ground target seeking and following scheme. Multiple real flight tests were conducted to verify the presented vision system. The experimental results show that this vision system is not only able to automatically detect and track the pre-defined ground target in the video sequence, but also able to guide the UAV to follow the motion of the target in-flight. The robustness and efficiency of the developed vision system for UAVs could be achieved by the current system. Our future research focus is to utilize the system for implementing vision-based automatic landing of the UAV on a moving platform in an environment without GPS signals.

## REFERENCES

- [1] M. J. Adriaans, R. C. Cribbs, B. T. Ingram and B. P. Gupta, "A160 Hummingbird unmanned air vehicle development program," *Proceedings of the AHS International Specialists' Meeting — Unmanned Rotorcraft: Design, Control and Testing*, Chandler, USA, 2007.
- [2] O. Amidi, T. Kanade and R. Miller, "Vision-based autonomous helicopter research at Carnegie Mellon Robotics Institute 1991–1997," *Proceedings of American Helicopter Society International Conference*, Gifu, Japan, pp. 1–12, 1998.
- [3] J. L. Barron, D. J. Fleet and S. S. Beauchemin, "Performance of optical flow techniques," *International Journal of Computer Vision*, vol. 12, pp. 43–77, 1994.
- [4] Y. Boykov and D. P. Huttenlocher, "Adaptive bayesian recognition in tracking rigid objects," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Hilton Head, SC, pp. 697–704, 2000.
- [5] G. R. Bradski and S. Clara, "Computer vision face tracking for use in a perceptual user interface," *Intel Technology Journal Q2 '98*, pp. 1–15, 1998.
- [6] G. W. Cai, B. M. Chen, K. M. Peng, M. B. Dong and T. H. Lee, "Modeling and Control of the Yaw Channel of a UAV Helicopter," *IEEE Transactions on Industrial Electronics*, vol. 55, pp. 3426–3434, 2008.
- [7] R. Canals, A. Roussel, J. L. Famechon and S. Treuillet, "A Biprocessor-Oriented Vision-Based Target Tracking System," *IEEE Transactions on Industrial Electronics*, vol. 49, pp. 500–506, 2002.
- [8] M. E. Campbell and W. W. Whitacre, "Cooperative tracking using vision measurements on seacraft UAVs," *IEEE Transactions on Control Systems Technology*, vol. 15, pp. 613–626, 2007.
- [9] F. Chaumette and S. Hutchinson, "Visual servo control part I: Basic approaches," *IEEE Robotics & Automation Magazine*, vol. 13, pp. 82–90, 2006.
- [10] F. Chaumette and S. Hutchinson, "Visual servo control part II: Advanced approaches," *IEEE Robotics & Automation Magazine*, vol. 14, pp. 109–118, 2007.
- [11] C. Chen, Y. F. Zheng, "Passive and active stereo vision for smooth surface detection of deformed plates," *IEEE Transactions on Industrial Electronics*, vol. 42, pp. 300–306, 1995.
- [12] Y. Cheng, "Mean Shift, Mode Seeking, and Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 790–799, 1995.
- [13] D. Comaniciu, V. Ramesh and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 564–577, 2003.
- [14] B. Enderle, "Commercial applications of UAVs in Japanese agriculture," *Proceedings of the AIAA 1st UAV Conference*, Portsmouth, Virginia, AIAA-2002-3400, 2002.
- [15] J. Ferruz, V. M. Vega, A. Ollero and V. Blanco, "Reconfigurable Control Architecture for Distributed Systems in the HERO Autonomous Helicopter," *IEEE Transactions on Industrial Electronics*, in press.
- [16] G. Franklin, J. D. Powell and A. E. Naeini, *Feedback Control of Dynamic Systems*, 4th edn. Upper Saddle River, New Jersey: Prentice-Hall, 2002.
- [17] N. Guenard, T. Hamel and R. Mahony, "A practical visual servo control for an unmanned aerial vehicle," *IEEE Transactions on Robotics*, vol. 24, pp. 331–340, 2008.
- [18] S. Hrbar, G. S. Sukhatme, P. Corke, K. Usher and J. Roberts, "Combined optic-flow and stereo-based navigation of urban canyons for a UAV," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3309–3316, 2005.
- [19] W. M. Hu, T. N. Tan, L. Wang and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 34, pp. 334–352, 2004.
- [20] Y. Hu, W. Zhao and L. Wang, "Vision-Based Target Tracking and Collision Avoidance for Two Autonomous Robotic Fish," *IEEE Transactions on Industrial Electronics*, vol. 56, pp. 1401–1410, 2009.
- [21] M. Isard and A. Blake, "Condensation conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, pp. 5–28, 1998.
- [22] E. N. Johnson, A. J. Calise, Y. Watanabe, J. Ha and J. C. Neidhoefer, "Real-time vision-based relative aircraft navigation," *Journal of Aerospace Computing, Information, and Communication*, vol. 4, pp. 707–738, 2007.
- [23] J. Kim and S. Sukkarieh, "SLAM aided GPS/INS navigation in GPS denied and unknown environments," *Proceedings of the 2004 International Symposium on GNSS/GPS*, Sydney, Australia, 2004.
- [24] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking, Part I: Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, pp. 1333–1364, 2003.
- [25] F. Lin, K. Y. Lum, B. M. Chen and T. H. Lee, "Development of a vision-based ground target detection and tracking system for a small unmanned helicopter," *Science in China — Series F: Information Sciences*, vol. 52, pp. 2201–2215, 2009.
- [26] B. Ludington, E. Johnso and G. Vachtsevanos, "Augmenting UAV autonomy," *IEEE Robotics & Automation Magazine*, vol. 13, pp. 63–71, 2006.
- [27] M. Meingast, C. Geyer and S. Sastry, "Vision based terrain recovery for landing unmanned aerial vehicles," *Proceedings of IEEE Conference on Decision and Control*, Atlantis, Bahamas, pp. 1670–1675, 2004.
- [28] L. Mejias, S. Saripalli, P. Cervera and G. S. Sukhatme, "Visual servoing of an autonomous helicopter in urban areas using feature tracking," *Journal of Field Robotics*, vol. 23, pp. 185–199, 2006.
- [29] R. Nelson and JR. Corby, "Machine Vision for Robotics," *IEEE Transactions on Industrial Electronics*, vol. IE-30, pp. 282–291, 1983.
- [30] F. Porikli, "Achieving real-time object detection and tracking under extreme conditions," *Journal of Real-Time Image Processing*, vol. 1, pp. 33–40, 2006.
- [31] F. Sadjadi, "Theory of invariant algebra and its use in automatic target recognition," *Physics of Automatic Target Recognition*, vol. 3, pp. 23–40, 2007.
- [32] Z. Sarris and S. Atlas, "Survey of UAV applications in civil markets," *Proceedings of the 9th Mediterranean Conference on Control and Automation*, vol. WA2-A, Dubrovnik, Croatia, pp. 1–11, 2001.
- [33] A. R. Smith, "Color gamut transform pairs," *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques*, New York, USA, pp. 12–19, 1978.
- [34] O. Spinka, O. Holub, and Z. Hanzalek, "Low-Cost Reconfigurable Control System for Small UAVs," *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 880–889, 2011.
- [35] T. Y. Tian, C. Tomasi and D. J. Heeger, "Comparison of Approaches to Egomotion Computation," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, pp. 315–320, 1996.
- [36] S. Tsugawa, "Vision-based vehicles in Japan: machine vision systems and driving control systems," *IEEE Transactions on Industrial Electronics*, vol. 41, pp. 398–405, 1994.
- [37] H. Veeraraghavan, P. Schrater and N. Papanikolopoulos, "Robust target detection and tracking through integration of motion, color and geometry," *Computer Vision and Image Understanding*, vol. 103, pp. 121–138, 2006.
- [38] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 780–785, 1997.
- [39] D. Xu, L. Han, M. Tan and Y. F. Li, "Ceiling-Based Visual Positioning for an Indoor Mobile Robot With Monocular Vision," *IEEE Transactions on Industrial Electronics*, vol. 56, pp. 1617–1628, 2009.
- [40] Q. M. Zhou, and J. K. Aggarwal, "Object tracking in an outdoor environment using fusion of features and cameras," *Image and Vision Computing*, vol. 24, pp. 1244–1255, 2006.



robot vision as well as embedded vision systems.

**Feng Lin** received the B.Eng. degree in Computer Science and Control, and the M.Eng. degree in system engineering from the Beihang University, Beijing, China, in 2000 and 2003, respectively. He received the Ph.D. degree in Computer & Electrical Engineering from the National University of Singapore in 2011. Currently, he is working as a Research Associate at Department of Electrical and Computer Engineering at National University of Singapore. His main research interests are unmanned aerial vehicles, vision-based control and navigation, target tracking,



**Xiangxu Dong** received the B.S. degree from the Department of Communications Engineering at Xiamen University in 2006. Since 2007, he has been a PhD candidate at National University of Singapore. His research interests include real-time software, cooperative coordination, formation control and unmanned aerial vehicles.



1993, he has been with Department of Electrical and Computer Engineering, National University of Singapore, where he is currently a professor. His current research interests are in robust control, systems theory, unmanned systems and financial market modeling.

He is the author/co-author of 8 research monographs including  $H_2$  Optimal Control (London: Prentice Hall, 1995);  $H_\infty$  Control and Its Applications (New York: Springer, 1998, Chinese edition published by Science Press, Beijing, 2010); Robust and  $H_\infty$  Control (New York: Springer, 2000); Linear Systems Theory (Boston: Birkhauser, 2004; Chinese translation published by Tsinghua University Press, 2008); Hard Disk Drive Servo Systems (New York: Springer, 1st Edition, 2002; 2nd Edition, 2006); and Unmanned Rotorcraft Systems (New York: Springer, 2011). He served/serves on the editorial boards for a number of international journals including IEEE Transactions on Automatic Control, Automatica, Systems & Control Letters, and Journal of Control Theory and Applications.

Dr Chen is a Fellow of IEEE. He was the recipient of Best Poster Paper Award, 2nd Asian Control Conference, Seoul, Korea (1997); University Researcher Award, National University of Singapore (2000); Prestigious Engineering Achievement Award, Institution of Engineers, Singapore (2001); Temasek Young Investigator Award, Defence Science & Technology Agency, Singapore (2003); Best Industrial Control Application Prize, 5th Asian Control Conference, Melbourne, Australia (2004); Best Application Paper Award, 7th Asian Control Conference, Hong Kong (2009); and Best Application Paper Award, 8th World Congress on Intelligent Control and Automation, Jinan, China (2010).



currently principal investigator of research programmes in control, guidance and multi-agent systems. His research interests include nonlinear dynamics and control, estimation and optimization, reduced-order modeling and control in aerodynamics. He is a member of IEEE Control Systems Society, and AIAA.

**Kai-Yew Lum** received his Diplôme d'Ingénieur from the Ecole Nationale Supérieure d'Ingénieurs Electriciens de Grenoble, France in 1988, and his M.Sc. and Ph.D. from the University of Michigan, Department of Aerospace Engineering, in 1995 and 1997, respectively. He worked in the DSO National Laboratories, Singapore as junior engineer from 1990 to 1993, then as senior engineer from 1998 to 2001, specializing in control and guidance technologies. He joined Temasek Laboratories at the National University of Singapore in 2001 where he is



Dr. Lee's research interests are in the areas of adaptive systems, knowledge-based control, intelligent mechatronics and computational intelligence. He currently holds Associate Editor appointments in the IEEE Transactions in Systems, Man and Cybernetics; IEEE Transactions in Industrial Electronics; Control Engineering Practice (an IFAC journal); and the International Journal of Systems Science (Taylor and Francis, London). In addition, he is the Deputy Editor-in-Chief of IFAC Mechatronics journal.

Dr. Lee was a recipient of the Cambridge University Charles Baker Prize in Engineering; the 2004 ASCC (Melbourne) Best Industrial Control Application Paper Prize; the 2009 IEEE ICMA Best Paper in Automation Prize; and the 2009 ASCC Best Application Paper Prize. He has also co-authored five research monographs (books), and holds four patents (two of which are in the technology area of adaptive systems, and the other two are in the area of intelligent mechatronics). He has published more than 300 international journal papers.

Dr. Lee was an Invited Panelist at the World Automation Congress, WAC2000 Maui U.S.A.; an Invited Keynote Speaker for IEEE International Symposium on Intelligent Control, IEEE ISIC 2003 Houston U.S.A.; an Invited Keynote Speaker for LSMS 2007, Shanghai China; an Invited Expert Panelist for IEEE AIM2009; an Invited Plenary Speaker for IASTED RTA 2009, Beijing China; an Invited Keynote Speaker for LSMS 2010, Shanghai China; and an Invited Keynote Speaker for IASTED CA 2010, Banff Canada.