<table>
<thead>
<tr>
<th><strong>Title</strong></th>
<th>Ground Stereo Vision-based Navigation for Autonomous Take-off and Landing of UAVs: A Chan-Vese Model Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Author(s)</strong></td>
<td>Tang, Dengqing; Hu, Tianjiang; Shen, Lincheng; Zhang, Daibing; Kong, Weiwei; Low, Kin Huat</td>
</tr>
<tr>
<td><strong>Date</strong></td>
<td>2016</td>
</tr>
<tr>
<td><strong>URL</strong></td>
<td><a href="http://hdl.handle.net/10220/41954">http://hdl.handle.net/10220/41954</a></td>
</tr>
<tr>
<td><strong>Rights</strong></td>
<td>© 2016 Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<a href="http://creativecommons.org/licenses/by/3.0">http://creativecommons.org/licenses/by/3.0</a>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.</td>
</tr>
</tbody>
</table>

Abstract

This article aims at flying target detection and localization of a fixed-wing unmanned aerial vehicle (UAV) autonomous take-off and landing within Global Navigation Satellite System (GNSS)-denied environments. A Chan-Vese model-based approach is proposed and developed for ground stereo vision detection. Extended Kalman Filter (EKF) is fused into state estimation to reduce the localization inaccuracy caused by measurement errors of object detection and Pan-Tilt unit (PTU) attitudes. Furthermore, the region-of-interest (ROI) setting up is conducted to improve the real-time capability. The present work contributes to real-time, accurate and robust features, compared with our previous works. Both offline and online experimental results validate the effectiveness and better performances of the proposed method against the traditional triangulation-based localization algorithm.

Keywords Localization, UAV, Chan-Vese Model, Ground Stereo Vision, EKF

1. Introduction

In the past few years, the integration of advanced robotic technologies with unmanned aerial vehicles (UAVs) opens new fields of applications. The opportunities and challenges of this fast-growing field were summarized by Kumar et al. [1]. Safe and autonomous motion control is shown definitely essential for various tasks. Within unknown or Global Navigation Satellite System (GNSS)-denied environments, autonomous navigation is an important component. Especially, autonomous take-off and landing has been of most importance as one of the most critical phases in many mission profiles of fixed-wing UAVs. Even small errors in guidance and control could yield system damages or even lost.

Besides GNSS and Inertial Measurement Unit (IMU), vision-based navigation has absorbed more attention from researchers in fixed-wing UAVs and quad-rotors as well. Both ground and onboard schemes have been developed for vision-based navigation. These onboard applications based on rotary-wing aircraft [2-4] and fixed-wing aircraft...
configure a single camera or multiple cameras on the aircrafts to detect runway or cooperative object for self-localization. Compared with onboard navigators, the ground system possesses stronger computation resources and saves cost because only one navigator is needed in each airport runway. Moreover, the ground scheme concentrates on autonomy of aerial vehicles’ take-off and landing on the runways.

Generally speaking, the ground vision-based autonomous take-off and landing has been mainly employed into fixed-wing vehicles [9]. LED-like visual fiducials are assigned onto UAVs to highlight the flying vehicles in images. Aiming at runway take-off, taxiing and landing of medium/large aerial vehicles, a ground stereo vision system has been developed and implemented in our previous works [10–15]. The system architecture and components are shown in Figure 1. Two cameras are symmetrically located on independent Pan-Tilt units (PTUs) to capture flying aircraft images. The PTU is controlled and feeds the pitching and yawing angles backward. Target detection of captured sequential images is accomplished by the specified computer. The three-dimensional localization algorithm is conducted for real-time online spatial estimation of the flying vehicle. In particular, the two PTUs are allocated on both sides of the airport runway to enlarge the baseline. It contributes to promoting the localization accuracy.

The ground stereo vision system involves several workflow steps, e.g. system calibration, image capture, target detection and localization, as shown in Figure 1. In this study, flying target detection and localization by using the captured sequential binocular images is emphasized. Tang et al. [10] initially employed the Chan-Vese model [16] into detecting the aircraft in images and then used triangulation to calculate the spatial coordinates. The experimental results were comparable with the corresponding differential GPS (DGPS) data in some sections. However, mismatching phenomena did exist within specified scenario, e.g., the aircraft flying out of field of view (FOV) of the cameras, and higher errors caused by automatic target detection. The real-time feature was unsatisfactory for practical applications since the average time consumption of the localization algorithm was about 157 ms, and the maximal even reached to be 212 ms.

Under such circumstances, this chapter concentrates on the above-mentioned problems, in terms of real-time capability, accuracy and robustness for the ground stereo vision system. The contributions can be stated as follows:

• An active region-of-interest (ROI) scheme is fused into the Chan-Vese model-based detection to improve the real-time feature of the proposed method. The present ROI is automatically generated according to predicted target position and size.

• A robust spatial localization method is developed by fusing Extended Kalman Filter (EKF) [17] into state estimation to reduce the localization inaccuracy caused by measurement errors of object detection and PTU attitudes.

This remainder is organized as follows. Section 2 provides a brief review of related works. Problem formulation on UAV autonomous take-off and landing is presented in Section 3. In Section 4, aircraft detection and spatial localization algorithms are developed and implemented. The advantages of localization precision, real-time capability and robustness are verified by experiments in Section 5. Finally, Section 6 concludes the work done.

2. Related Works

The ground stereo vision navigation is developed for UAV autonomous take-off and landing within GNSS-denied...
environments. Flying aircraft detection is emphasized since it is one kernel component of the ground vision system. To our best knowledge, both corner-based and skeleton-based algorithms were employed into the flying object detection on the ground captured sequential images. In this study, one of skeleton-based schemes are concerned, and active contour is applied into extracting skeletons or edges in the images.

2.1 Active contour object detection

Active contour, whose basic idea is evolving a curve according to a partial differential equation (PDE), is constructed to extract the contour of an object. Active contour was introduced by Kass et al. [18] to segment objects in images using dynamic curves. As developed in [19], the active contour is further divided into parametric active contour models and geometric active contour models. The parametric active contours \cite{18, 20} are expressed as the parameterized curves and optimally evolve in a Lagrangian framework. They have been used extensively in many applications over the last decades. The geometric active contours, introduced by Caselles et al. \cite{21} and Malladi et al. \cite{22}, are represented implicitly as level sets of a two-dimensional distance functions that evolve according to an Eulerian framework.

To our knowledge, the image segmentation has played an important role in many applications like object tracking, medical image analysis \cite{23}, etc. As a category of classical image segmentation methods, the active contour model, however, has some problems in energy minimization process (described in Section 4.1) including initialization, stopping functions design and anti-noise robustness. To overcome some of these difficulties, Ksantini et al. \cite{24} proposed a new active contour model that completely eliminated the need of the re-initialization procedure. Moreover, a polarity-based stopping function was proposed to automatically end the curve evolution instead of the widely used gradient-based stopping function. More recently, a novel reaction-diffusion method for implicit active contours was developed, which outperforms other classical active contour methods on noisy image in work presented in \cite{25}.

Gradient Vector Flow (GVF) \cite{20} is a modified Snake algorithm. It solves two key problems: the first is that Snakes cannot move toward objects that are too far away and the second is that the Snakes cannot move into boundary concavities or indentations (such as the top of the character U). Chan-Vese model proposed by Chan and Vese \cite{16} is able to automatically detect all contours regardless of the location of the initial contour. In this chapter, it is employed and developed to detect the flying target. In Section 5, we will compare the Chan-Vese model with GVF and Snake in terms of the detection performances.

2.2 Vision-based autonomous landing

Unlike GNSS-based navigation system \cite{26}, vision-based navigator is designed to achieve a successful GNSS-denied flight. Considering solving the landing problem, most researchers focus on onboard vision, such as a novel two-camera method for estimating the full six-degree-of-freedom pose of the helicopter in work by E. Altug et al. \cite{27} and fixing a forward camera on a UAV to detect lines of target runway \cite{28}. On the other hand, ground vision-based navigation systems have seldom considered for autonomous landing. In several ground vision-based navigation systems, Martinez et al. \cite{29, 30} estimated the helicopter’s position based on the detection and tracking of planar structures, but it relied on artificial markers. Another ground vision-based system used one CCD camera to recognize a square marker which size is already known and then measure the three-dimensional coordinates of the flight body, while the limited ranges of observation restricted the take-off and landing range \cite{31}.

3. Problem Formulation

The ground stereo vision-based localization algorithm is to discern where the UAV object is during its autonomous take-off and landing process. Digital images and camera attitudes are the input of this algorithm and the real-time UAV position is the output. The whole algorithm is composed of two sections: Chan-Vese model-based object detection and EKF-based object localization. Object detection operator $FE(\cdot)$ and localization operator $FL(\cdot)$ are defined as follows:

$$FE(\cdot): I \rightarrow \text{Pos}_I$$

$$FL(\cdot): \{\text{Pos}_I, C_{\text{PTU}}\} \rightarrow \text{Pos}_S$$

where $I$ is digital image domain, $\text{Pos}_I$ denotes target position in image (unit: pixel), and $\text{Pos}_S$ denotes target spatial position (unit: meter). As shown in Figure 2, the object detection operator $FE(\cdot)$ includes UAV Detection module, Initialization module and UAV Trajectory Estimation module. UAV Detection module mainly contains three components: region segmentation operator, region filtering operator and centering operator described in a previous work \cite{10}. Object localization operator $FL(\cdot)$ is corresponding to UAV Spatial Position module.

3.1 Chan-Vese model-based object detection operator $FE(\cdot)$

Compared with the previous work \cite{10}, this chapter mainly improves the Initialization module for better performances on real-time capability. In previous work, we modeled the trajectory as a piecewise straight line. The estimated spatial position $\text{Pos}_S^k$ at step $k$ is predicted by former position at step $k-1$ and $k-2$:

$$\text{Pos}_S^k = (X_k, Y_k, Z_k) = 2\text{Pos}_S^{k-1} - \text{Pos}_S^{k-2} \quad (1)$$
It is assumed that the target trajectory is with continuous first-order derivative, which is more realistic than previous assumption of piecewise linearity of motion [10]. The trajectory \( m(t) \) can be described as:

\[
\lim_{t \to \infty} m'(t) = \lim_{t \to \infty} m'(t_0) \quad \forall t_0 \in (t_{\text{start}}, t_{\text{end}}) \tag{2}
\]

To realize smooth first-order differential of trajectory, modeling parametric curve piecewise is feasible. It is realized with recursion that is to say the trajectory is renewed by adding a new parametric curve circularly at each step. The detail will be discussed in Section 4.

In the present work, \( \text{Pos}_{k-1}^L \), \( \text{Pos}_{k-2}^L \) and target speed \( \text{V}_{k-1}^L \) at step \( k-1 \) are utilized to estimate target spatial position at step \( k \). Define spatial position estimation operator \( \text{EST}_p(\cdot) \):

\[
\{ \text{Pos}^L_{k-2}, \text{Pos}^L_{k-1}, \text{V}^L_{k-1} \} \rightarrow \text{Tra}_k
\]

\[
\text{Tra}_k \rightarrow (X_k, Y_k, Z_k)
\]

where \( \text{Tra}_k \) denotes the new parametric curve estimated in step \( k \). The estimated spatial position is calculated by \( \text{Tra}_k \), \( \text{EST}_p(\cdot) \), which is presented in Figure 2.

3.2 EKF-based spatial localization operator \( FL(\cdot) \)

With strong robustness, real-time capability and little memory cost, Kalman Filter (KF) [32] can estimate the system state by the measurement. The KF uses a series of measurements containing streams of noise and other inaccuracies to produce a more accurate estimation of unknown variables than those based on a single measurement alone. In other words, the KF operates recursively on streams of noisy input data to estimate a statistically optimal system state.

In this chapter, object spatial position and speed, PTU attitude and rotation velocity are modeled as the state \( x \) estimated by EKF for the nonlinear measurement model:

\[
x = [\text{Pos}^L, \text{V}^L, C^w, C^r] \tag{3}
\]

where all components are expressed in the global coordinate and they can be written as:

\[
\text{Pos}^L = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^	op
\]

\[
\text{V}^L = \begin{bmatrix} v_x & v_y & v_z \end{bmatrix}^	op
\]

\[
C^w = [\text{lp} & \text{li} & \text{rp} & \text{ri}]
\]

\[
C^r = [\text{wl} & \text{wl} & \text{wr} & \text{wr}]
\tag{4}
\]

where \( \text{lp} \) and \( \text{li} \) denote left camera’s yaw and pitch angle, and \( \text{rp} \) and \( \text{ri} \) are the same to right camera. The \( \text{wl} \), \( \text{wr} \), \( \text{wl} \) and \( \text{wr} \) are corresponding angular velocity.

The measurement \( z \) is modeled as follows:

\[
z = [\text{Pos}^L, \text{Pos}^L, \text{PTU}^w] \tag{5}
\]

where \( \text{Pos}^L \) and \( \text{Pos}^L \) denote target image positions in left \( (u^L_i, v^L_i) \) and right images \( (u^R_i, v^R_i) \) respectively. \( \text{PTU}^w \) represents left and right PTUs attitudes, namely yaw and pitch angles. Meanwhile, the zero-mean Gaussian noise is modeled in the process and measurement models. The details are described in Section 4.2.

4. Methodology

The whole localization algorithm is based on ground stereo vision system. The constant parameters are obtained by calibration, described in previous work [14], such as camera intrinsic and external parameters, PTU position and initial attitude, etc. Also, images captured from two cameras and PTUs attitude can be obtained in real time. With this information, extracting object using Chan-Vese model-based detection algorithm in image firstly and then locating the object with EKF-based spatial localization algorithm are done.

4.1 Chan-Vese model-based object detection algorithm

Chan-Vese model is a kind of geometry active contour model related with curve evolution theory and level set method introduced by Osher and Sethian for capturing
moving fronts [33]. Level set method increases the problem's dimension to be higher. For example, a plane curve $C$ is implicitly expressed as a same-value curve of threedimensional continuous functional surface $\phi(x, y, t)$, which is called level set function. The active contour can be expressed as zero level set of level set function indirectly. This expression leads to its adaption to topology structure change, and a stable numerical algorithm.

The Chan-Vese detection algorithm aims at finding the target boundary by using an active contour approach. The active contour is dynamically evolved by minimizing the value of energy function $F(C)$ with the level set method. As shown in Figure 3, an active contour is initially given and then it evaluates as the iteration of solving numerical partial differential equations. Finally, it will converge to the target boundary. The energy function $F(C)$ in Chan-Vese model is as follows:

$$F(C) = \mu L(C) + \text{varea}(\text{inside}C)$$

$$+ \lambda \int_{\text{inside}} (u-c_1)^2 \, dx \, dy$$

$$+ \lambda \int_{\text{outside}} (u-c_2)^2 \, dx \, dy$$

where $C$ is the ranging closed curve, and $u$ is the matrix of the image. $\mu$ and $\nu$ are coefficients. $c_1$ and $c_2$ are average pixel intensity values of inside and outside regions of the closed curve respectively. Therefore $(u-c_1)^2$ and $(u-c_2)^2$ can be treated as the pixel intensity values' variance matrices of inside and outside regions. $L(C)$ is the length of closed curve.

From the viewpoint of applications, the schematic workflow of Chan-Vese detection algorithm is shown in Figure 4. This standard Chan-Vese algorithm has been employed into the flying aircraft detection with higher time cost [10]. Eventually, a ROI Scheme (red rectangle) is fused into the Chan-Vese model-based detection to improve its real-time capability promotion then.

The computation time is undoubtedly reduced by focusing on the smaller-scale ROI, but it should be ensured that the aircraft is within the ROI. The detection accuracy on the target becomes an important issue. Thereafter, a more rational and precise trajectory should be estimated. It was assumed that the target is of piecewise linearity of motion previously and estimated current target spatial position as (1). As mentioned in Section 3, the speed of target is smoothed according to (2) and the target is supposed to move along with the trajectory with smooth first-order differential. These assumptions regarding object motion are more practical, and that the use of these assumptions will improve position estimation accuracy.

Assuming that the target has been located before step $k$, and spatial positions are marked as $P_{S,i-1}$ and $P_{S,i-2}$ at step $k-1$ and $k-2$. The speed is represented as $V_{S,i}^{k-1}$ and $V_{S,i}^{k-2}$. Within the plane composed of them, calculating a parametric curve equation satisfied with two conditions:

- The points $P_{S,i-1}$ and $P_{S,i-2}$ are on the parametric curve.
- The tangential direction is the same as the direction of $V_{S,i}^{k-2}$.

The parametric curve $G_i(x, y, z)=0$ can be expressed as:

$$G_i(P_{S,i-1}) = 0$$

$$G_i(P_{S,i-2}) = 0$$

$$\Rightarrow G_i(x, y, z) = 0$$

$$\text{grad}(G_i(P_{S,i-1}))|V_{S,i}^{k-1} = 1$$

where $\text{grad}(G_i(P_{S,i-2}))$ denotes the tangent vector at point $P_{S,i-2}$ and $V_{S,i}^{k-1}$ is the tangent vector at point $P_{S,i-1}$ after the parametric curve is obtained.
Supposing that the target kept moving with a constant speed (possibly different direction) during step \( k-1 \) and \( k \), one point is obtained on this parametric curve:

\[
d | \overrightarrow{Pos^s_{k-1}} \overrightarrow{Pos^s_{k}} | = d | \overrightarrow{Pos^s_{k-1}} \overrightarrow{Pos^s_{k-2}} |
\]

where \( \overrightarrow{Pos^s_{k}} \) represents estimated spatial position at current frame. \( \overrightarrow{Pos^s_{k}} \) is projected into image with camera external parameters acquired from PTU directly as the center of \( \phi_0(x, y) = 0 \) and ROI.

### 4.2 EKF-based object spatial localization algorithm

Spatial localization is the following step of flying aircraft detection. Tang et al. [10] pointed out that the localization precision is significantly affected by both target detection errors and PTU attitude errors. High detection errors occur when most of the aircraft body is flying out of the field of view. Typically, the localization error was up to be about 29.12 m in the worst occasion.

Under such circumstances, (EKF) is employed to estimate on the aircraft spatial coordinates and suppresses the measurement error influence to localization accuracy. In this study, the EKF is used to estimate detected target positions in images and binocular camera attitudes, and the state \( x \) is modeled as Equations (3) and (4).

#### 4.2.1 EKF Prediction

Let us estimate the state at step \( k \) by the state at step \( k-1 \):

\[
\overrightarrow{x_{k|k-1}} = F_{k} \overrightarrow{x_{k-1|k-1}}
\]

where \( F_{k} \) is a state transition matrix. Object position and camera attitude are predicted as follows:

\[
\overrightarrow{Pos_{k|k-1}} = V^s_{k-1} \Delta t + \overrightarrow{Pos_{k-1|k-1}}
\]

\[
\overrightarrow{C_{k|k-1}} = C^u_{k-1} \Delta t + \overrightarrow{C_{k-1|k-1}}
\]

where \( \Delta t \) represents the time interval between two neighboring continuous frames. The state transition matrix can be written as:

\[
\begin{bmatrix}
I_{3x3} & \Delta t_{3x3} & 0_{3x4} & 0_{3x4} \\
0_{3x3} & I_{3x3} & 0_{3x4} & 0_{3x4} \\
0_{4x3} & 0_{4x3} & I_{4x4} & \Delta t_{4x4} \\
0_{4x3} & 0_{4x3} & 0_{4x4} & I_{4x4}
\end{bmatrix}
\]

#### 4.2.2 EKF measurement model

Equation (5) presents the measurement \( z \), which is composed of three components: \( Pos^I_{k-1} \), \( Pos^I_{k} \) and \( PTU^{At} \).

Figure 5. The pinhole model of camera. \( P(x_0,y_0,z_0) \) is the spatial point and \( p \) is the projection in image plane.

Pinhole camera model is used to project the estimated target position \( \overrightarrow{Pos^I_{k-1}} \) to the image as depicted in Figure 5. It should be noted that the cameras are configured closely to PTUs rotation shafts, so that the camera positions hardly change and they are regarded as constants in this chapter. The estimated camera attitudes \( \overrightarrow{C_{At}^{Ii}} \) and calibration parameters including camera positions and camera intrinsic parameters are necessary to solve target images positions \( Pos^I \):

\[
\begin{bmatrix}
\overrightarrow{Pos^I_{k}} \\
\overrightarrow{C_{At}^{Ii}_{k}}
\end{bmatrix} = \lambda N \overrightarrow{Pos^I_{k-1}}
\]

where \( \Delta t_{n \times n} \) stands for \( n \times n \) matrix, whose diagonal elements are \( \Delta t \) and the others are 0. Estimate state covariance matrix at step \( k \):

\[
P_{k|k-1} = F_{k} P_{k-1|k-1} F_{k}^T + G_{k} Q_{k} G_{k}^T
\]

where \( Q_{k} \) is covariance of the dynamic model with a zero-mean Gaussian noise, and \( G_{k} \) is the Jacobian matrix of \( Q_{k} \) with respect to the state \( x \) at step \( k \). The state covariance matrix can be divided into four parts: UAV to UAV, UAV to Camera, Camera to Camera and Camera to UAV:

\[
P_{i} = \begin{bmatrix}
P_{u/u} & P_{u/c} \\
P_{c/u} & P_{c/c}
\end{bmatrix}
\]
where $\lambda$ is normalization coefficient. $T_{k|k-1}^{\text{out}}$ and $N_{k|k-1}^\text{in}$ are transformation matrixes depended on camera external and intrinsic parameters:

$$
\begin{bmatrix}
\tilde{T}_{k|k-1}^\text{out} & \tilde{t}
\end{bmatrix} =
\begin{bmatrix}
\tilde{R}_{k|k-1} & \tilde{t}
\end{bmatrix}
$$

$$
N_{k|k}^\text{in} =
\begin{bmatrix}
1 / d_i & 0 & u_i & 0 & f & 0 & 0 \\
0 & 1 / d_i & v_i & 0 & f & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

where $\tilde{R}_{k|k-1}^C$ is the rotation matrix of $\tilde{C}_{k|k-1}$. $t$ represents the translation vector between camera coordinate and global coordinate.

Next, the measurement model is written as:

$$
\begin{bmatrix}
\tilde{z}_k
\end{bmatrix} =
\begin{bmatrix}
\tilde{\text{POS}}_k^i \\
\tilde{\text{POS}}_k^s \\
\tilde{\text{PTU}}_k^i
\end{bmatrix} =
J(\lambda, N_{k|k}^\text{in}, \tilde{T}_{k|k-1}^{\text{out}}, \tilde{\text{POS}}_{k|k-1}^s) \begin{bmatrix}
\tilde{\text{POS}}_k^i \\
\tilde{\text{POS}}_k^s \\
\tilde{\text{PTU}}_k^i
\end{bmatrix}
$$

where $J(\cdot)$ outputs a new vector containing the first two components of the input vector.

In some cases, the estimated position is out of FOV, which means that the target projection is not in the images. Thus, the untrusted detection results should be corrected. Intuitively understanding, the further the estimated target image position is away from image boundary, the smaller the probability is that the real target is within the image. A coefficient $T$ is introduced to weigh the probability that the target is within the image, and the target image position $(u_k, v_k)$ is updated by $T$:

$$(u_k, v_k) = (1 - T_k)(u_k, v_k) + T_k(\tilde{u}_k, \tilde{v}_k) \quad (7)$$

where $T_k \in [0, 1]$ which is calculated as

$$
T_k = \begin{cases} 
0, & (\tilde{u}_k, \tilde{v}_k) \in I \\
\frac{d_k^i}{d_k^i + l(d_k^e - d_k^i)}, & \text{else} 
\end{cases} \quad (8)
$$

where $I$ is digital image domain, $d_k^i$ denotes pixel distance between estimated target image position and image center and the part of it beyond image is $d_k^e$, and $l$ is a coefficient within the range $[0, 1]$.

### 4.2.3 EKF correction model

The $x_k$ is updated by measurement $z_k$ and covariance matrixes. According to the measurement model $h$, its Jacobian matrix $H_k$ is described as:

$$
H_k = \begin{bmatrix}
\frac{\partial h(\tilde{\text{POS}}_{k|k-1}^s, \tilde{\text{POS}}_{k|k-1}^e)}{\partial x_k}
\end{bmatrix}
$$

The Kalman gain $K_k$ is:

$$
S_k = H_k P_{k|k-1}^{x_k} H_k^T + R
$$

$$
K_k = P_{k|k-1}^{x_k} H_k^T (S_k)^{-1}
$$

where $R$ is Gaussian noise covariance matrix of the sensor for each measurement. Update state and covariance matrix:

$$
\begin{bmatrix}
\tilde{x}_{k|k-1} \\
P_{k|k-1}^{x_k}
\end{bmatrix} =
\begin{bmatrix}
\tilde{x}_{k|k-1} \\
P_{k|k-1}^{x_k}
\end{bmatrix} +
K_k (x_k - h(\tilde{x}_{k|k-1}))
$$

$$
P_{k|k}^{x_k} = (1 - K_k H_k) P_{k|k-1}^{x_k}
$$

Figure 6 shows the flow chart of the EKF-based localization operator $FL(\cdot)$. First, state $\tilde{x}_{1|k-1}$ is predicted. The target spatial position is then projected into image and checkout whether it is beyond image. If it is beyond image, the measurement should be updated according to (7). Finally the new state $x$ is achieved by EKF correction procedure.
Algorithm 1 Ground Stereo Vision-Based Localization

Inputs:
- Captured images \( I \).
- Two PTU attitudes.
- Previous state \( x_{k-1} \).
- Predicted center and radius of \( \phi_{\theta}(x, y) = 0 \).
- Predicted center, height and width of ROI.
- Transformations between coordinate systems.

Output: Current state \( x_k \).

Procedure:
Step 1: Image object detection \( FE() \):
  a) Image preprocessing: The \( S \) channel of HSV model is obtained as input image of Chan-Vese model-based segmentation.
  b) ROI set: ROI is set up according to predicted ROI parameters.
  c) Region segmentation: \( \phi_{\theta}(x, y) = 0 \) is initialized, and segmenting several target-like regions.
  d) Image localization: Filtering non-object regions and locating target center in image.

Step 2: Object localization \( FL() \):
  a) State estimation: Estimating state \( \bar{x} \) by (6).
  b) Measurement updating: Determining whether the estimated target is out of the FOV, and updating \( z_k \) by (7).
  c) State correction: Calculating Kalman gain \( K \), and correcting estimated state \( \bar{x} \).

Step 3: Parameter prediction: Estimating target trajectory and predicting parameters of \( \phi_{\theta}(x, y) = 0 \) and ROI for next frame.

5. Experiments and Discussion

UAV autonomous take-off and landing system based on stereo vision is built for the experiments of this object localization algorithm [11–13]. As shown in Figure 1, the precise PTUs are set up on two sides of the runway, and the baseline is about 10.77 m. The visible light camera DFK 23G445 is set up on the PTU and turn along with PTU to extend field of view. At the same time, the high position resolution (0.00625 degree) and high rotatory speed (50 degree/sec) of PTU enables higher accuracy localization. Figure 1 also presents the aircraft for experiments, whose wingspan is approximately 2.3 m.

The proposed approach is to be validated by both offline and online experiments. The offline experimental data, including calibration parameters, videos, PTU attitude, and synchronous DGPS data, are obtained during autonomous take-off and landing guided by DGPS. As the product manual of DGPS lists, the localization error of DGPS is less than 2 cm 95% of the time. These experiments mainly verify four performances improvements compared with previous works and other state-of-the-arts:

- First experiment confirms the better accuracy of Chan-Vese model-based object detection algorithm related to other active contours algorithms, such as snake and GVF snake.
- Second experiment shows the real-time capability improvement of Chan-Vese model-based object detection algorithm.
- Third experiment validates higher localization accuracy with EKF.
- Final experiment shows better robustness to measurement error with EKF.

The center point of aircraft head is treated as target image coordinate in these experiments. Note that the online experiments are implemented during UAV autonomous take-off and landing, but the results are not offered to UAV as localization information for autonomous take-off and landing considering its safety. It should be noted that the algorithms groups we used in experiments are marked as follows:

- CV_E denotes Chan-Vese model-based detection algorithm and EKF-based localization algorithm.
- CV_T denotes Chan-Vese model-based detection algorithm and triangulation-based localization algorithm.

5.1 Object detection experiments

Object detection in image with high accuracy is the foundation to obtain highly accurate localization from stereo vision system. At the same time, real-time processing capability is the required key factor in task implementation.

5.1.1 Chan-Vese model-based detection accuracy experiments

To validate the high UAV detection accuracy using Chan-Vese model, we compared this Chan-Vese model-based object detection algorithm with Snake and GVF algorithms in terms of detection accuracy.

Figure 7 shows the segmentation procedures and results with Snake-based and Chan-Vese model-based algorithms respectively. The pre-processing is the histogram equalization result of \( S \) channel of HSV model. Experimental results show that the aircraft in image often topologically changes after the image pre-processing. The wing of UAV disconnects with main body after preprocessing (Figure 7). The segmentation result of Snake-based algorithm surrounded by red contour does not contain the wing because Snake is not adapted to topological change, which may lose integral target information and lead to detection error.
Figure 7. Comparison of Snake-based with Chan-Vese model-based region segmentation in terms of adaptability to topological change. The two right images are corresponding to the region surrounded by the yellow contour and they show the segmentation results encircled by the red contours with Snake and Chan-Vese algorithms, respectively.

One set of offline data collected from an autonomous landing are used to compare Snake algorithm and GVF with Chan-Vese model in terms of detection accuracy. The red, light blue and green trajectories in Figure 8 are generated by different detection algorithm but same localization algorithm (triangulation). Blue is generated by DGPS, and red and light blue are generated by GVF and Snake algorithm, and green is generated by Chan-Vese algorithm. When the UAV is closed to ground, which means that the background is most complex, the GVF and Snake algorithm-based localization errors increase. Images and localization results in points G and H are picked up and showed in Figure 9. It shows that the detection error is mainly caused by the wrong aircraft region segmentation because GVF and Snake algorithm can only segment one region. Chan-Vese algorithm segments several regions including the target region and picks up the right one through region filtering.

5.1.2 Chan-Vese model-based detection real-time capability experiments

The real-time capability limit of the detection algorithm has been pointed out, which reaches about 157 ms every frame. The time consumption test consequence is that the Chan-Vese model-based region segmentation takes almost 84.1% of the total time required to process a frame of video. Hence, the reduction of region segmentation time consumption should be a most effective way to promote real-time capability.

(a) UAV autonomous landing trajectories generated by different object detection algorithm. Real images are captured from left and right cameras in three positions of the landing trajectory.

(b) Trajectory projections in XOY and ZOY planes.

Figure 8. Localization results using different object detection algorithm. The blue trajectory is the reference trajectory generated by DGPS. The red, light blue and green trajectories are generated by GVF, Snake and Chan-Vese detection algorithm and triangulation-based localization algorithm. The Y global axis is along with the direction of the runway.
The real-time capability is tested in the same equipment in the previous work [10], which is a PC with 2.80 GHZ CPU and 6.00 GB RAM. Figure 10 indicates the reduction of segmentation time consumption with ROI setting. The time consumption decreases significantly as the ROI size reduces. However, the detection accuracy should not be ignored. Table 1 shows the detection average error at each image coordinate. With comprehensive consideration, 250*100 (250 is the width and 100 is the height) is the best basic ROI size, whose average errors are all below 1.00 pixel and average time consumption is about 20 ms. Most importantly, the time consumption of 250*100 in one video distributes relatively concentrative, which is a significant performance in real project.

We compare the average time consumption of object segmentations with different algorithms, and these segmentations are with same initial conditions, iteration stop conditions and other conditions. The results are shown in Table 2. The Snake algorithm performs best but compared with Chan-Vese and GVF algorithms, Snake shows little advantage.

5.2 EKF-based localization experiments

Triangulation was used in previous work to locate the aircraft. Experiment results indicate that its localization accuracy and robustness are deeply affected by measurement accuracy including object detection and PTU attitude. To reduce the influence of measurement error effectively, EKF is introduced to estimate UAV position. Accuracy and robustness are improved in offline and online experiments.

5.2.1 Localization accuracy experiments

DGPS data are collected during autonomous landing process. These experiments treat them as the reference
localization results. Two set of landing data are chosen to present the accuracy improvement with EKF. It should be noted that the UAV position and PTUs attitudes in EKF initial state are configured by DGPS and PTUs measurements because the UAV is located by DGPS before its autonomous landing.

The offline experiments discussed above are all about autonomous landing process, which is guided by DGPS for position and IMU for attitude. Figure 11 shows autonomous landing trajectories and localization errors in X, Y and Z directions. The blue trajectory is generated by DGPS, and the green and red ones are generated by CV_T and CV_E algorithms respectively. The three shadow areas denote three landing periods in which the image backgrounds are only sky, sky and ground and only ground, respectively. Table 3 indicates the root-mean-square error (RMSE) and improvement of each axis using EKF.

There is a delay between images and DGPS for out-sync of data transmission, so it should be taken in consideration when the deviation is calculated. Assuming that the delay is constant and it is tested by experiments. The constant delay is set up to three image frames. In Figure 11(b), the deviation increases fleetly when the y coordinate is less than 187. That is caused by the weak DGPS signal when the aircraft is closed to ground. Through a set of electromagnetic environment test, the conclusion is that the electromagnetic environment is most complex in the region closed to the ground (less than 2 m), which leads to the weakest DGPS signal. When the DGPS signal is absent, the DGPS data are kept constant. Furthermore, big delays caused by continuous DGPS signal interruption are accumulated. That is why the deviation (y < 187 m) (Table 3) becomes bigger as image number increases.

In Table 3, EKF-based localization algorithm shows better accuracy especially in Z axis. Also, accuracy at X axis is improved obviously. And the deviation at Y axis is still about 2–3 m. The accuracy of height is most effective to real UAV autonomous landing. On the contrary, the accuracy in Y axis has minimal influence on its autonomous landing because of the long runway (about 400 m). The EKF-based localization algorithm reduces the deviation at Z axis for about 67.16% and 77.75%, which shows a significant improvement to autonomous landing. The limited width (15 m) of runway also needs a high localization accuracy at X axis. In conclusion, the localization accuracy improvement brought by EKF-based localization algorithm shows great significance to UAV autonomous landing.

In Table 3, RMSE of second experiment is calculated only from the trajectory whose y coordinate is bigger than 187 because of the incorrect DGPS data. Table 3 compares the errors in the X and Z directions, while the errors in Y direction (runway direction) are clearly higher. Figure 12 shows the localization errors caused by object detection errors (0-50 pixels) in u direction and v direction of image. Obviously, the detection errors cause significantly higher
localization errors in y direction, because the baseline is much shorter than runway, and according to camera pinhole model (see Figure 5), an angle error will lead to higher localization error in runway direction.

5.2.2 Localization robustness experiments

When the predicted target image coordinates \( (\tilde{u}_k, \tilde{v}_k) \) is out of the FOV at some points, As given in (7) and (8), the measurements are corrected by coefficient \( T_k \). Figure 13 shows several correction results with coefficient \( T_k \). Figures 13(a), (b), (c) and (d) are corresponding to A, B, C and D in Figure 11(a). Blue points are original target image coordinates extracted by Chan-Vese model-based object detection algorithm and red points are correctional target image coordinates.

In Figure 11(b), the trajectory generated by CV_T in region M deviates the reference trajectory obviously. It is caused by object detection error. EKF-based localization algorithm is not so sensitive to measurement error for the use of historical information. Table 4 shows the localization RMSE in region M at each axis.

The object detection error can be weighed by localization deviation with triangulation algorithm. The great promotion of localization accuracy at Z axis makes the impossible autonomous landing with 1.3028 m deviation to be realizable with 0.2111 m deviation. The results indicate the good robustness of EKF-based localization algorithm to measurement error.

5.3 Online localization experiments

The online localization experiments include an autonomous take-off and an autonomous landing. For the safety of UAV, it is still located by DGPS, and this localization algorithm is working in the ground station at the same time. The localization results are transmitted from ground computer to the UAV by the radio stations configured on ground and UAV. The transmission time delay is related to the transmission distance. We have tested the transmission delay and the results show that during autonomous landing it is about 30 ms. With additional images transmission and processing time consumption (about 80 ms), the total time delay is about 110 ms. According to the estimated UAV speed, we compensate for the final localization results assuming that the UAV is with constant speed during this time delay.

---

**Table 3.** The root-mean-square error (RMSE) and precision improvement with EKF at each axis. Only the trajectory \( y>187 \) m is used to calculate the RMSE in index 2.

<table>
<thead>
<tr>
<th>Index</th>
<th>Algorithm</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CV_T (m)</td>
<td>0.3166</td>
<td>2.9019</td>
<td>0.3873</td>
</tr>
<tr>
<td></td>
<td>CV_E (m)</td>
<td>0.3104</td>
<td>2.7920</td>
<td>0.1272</td>
</tr>
<tr>
<td></td>
<td>Improvement</td>
<td>1.96%</td>
<td>3.79%</td>
<td>67.16%</td>
</tr>
<tr>
<td>2</td>
<td>CV_T (m)</td>
<td>0.2728</td>
<td>2.4367</td>
<td>0.5393</td>
</tr>
<tr>
<td>(&lt;187 m)</td>
<td>CV_E (m)</td>
<td>0.2708</td>
<td>1.8114</td>
<td>0.1200</td>
</tr>
<tr>
<td></td>
<td>Improvement</td>
<td>0.73%</td>
<td>25.66%</td>
<td>77.75%</td>
</tr>
</tbody>
</table>

**Table 4.** The RMSE and precision improvement with EKF of region M in Figure 11(b)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>CV_T (m)</td>
<td>0.1967</td>
<td>1.7565</td>
<td>1.3028</td>
</tr>
<tr>
<td>CV_E (m)</td>
<td>0.1832</td>
<td>1.5914</td>
<td>0.2111</td>
</tr>
<tr>
<td>Improvement</td>
<td>6.86%</td>
<td>9.40%</td>
<td>83.80%</td>
</tr>
</tbody>
</table>

---

![Figure 12](image1.png) Localization errors in X, Y and Z axis directions caused by different object image position errors. (a) The localization errors caused by equal errors in \( u \) and \( v \) directions. (b) Constant error in \( v \) direction and changeable error in \( u \) direction. (c) Constant error in \( u \) direction and changeable error in \( v \) direction.

![Figure 13](image2.png) The object detection and correction results in four special images respectively. (a) Detection and correction results in point A. (b) Detection and correction results in point B. (c) Detection and correction results in point C. (d) Detection and correction results in point D. These points are marked in Figure 11 (a). The projections of predicted UAV positions in the four images are out of FOV.

![Figure 14](image3.png) Localization errors in X, Y and Z axis directions caused by different object image position errors. (a) The localization errors caused by equal errors in \( u \) and \( v \) directions. (b) Constant error in \( v \) direction and changeable error in \( u \) direction. (c) Constant error in \( u \) direction and changeable error in \( v \) direction.
Figure 14 shows the localization results. The red trajectory is generated by this localization algorithm (CV_E) online, and the green trajectory is generated by CV_T algorithm with offline data for the accuracy comparison. The whole algorithm consumes about 100 ms of every frame. It means that the localization frequency is about 10 HZ which is available for real autonomous take-off and landing. Table 5 is the localization RMSE comparison of online experiments. The CV_E algorithm also performs better as offline experiments show. The RMSE at Y axis of take-off experiments reaches 15.1831 m, but it also meets the autonomous take-off demand owing to the long runway.

<table>
<thead>
<tr>
<th>Experiments</th>
<th>Algorithm</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>Take-off (RMSE)</td>
<td>CV_T (m)</td>
<td>0.4349</td>
<td>15.2443</td>
<td>1.0405</td>
</tr>
<tr>
<td></td>
<td>CV_E (m)</td>
<td>0.4341</td>
<td>15.1831</td>
<td>0.3122</td>
</tr>
<tr>
<td></td>
<td>Improvement</td>
<td>0.18%</td>
<td>0.41%</td>
<td>70.00%</td>
</tr>
<tr>
<td>Landing (RMSE)</td>
<td>CV_T (m)</td>
<td>0.1401</td>
<td>1.8303</td>
<td>0.5477</td>
</tr>
<tr>
<td></td>
<td>CV_E (m)</td>
<td>0.1368</td>
<td>1.7262</td>
<td>0.1291</td>
</tr>
<tr>
<td></td>
<td>Improvement</td>
<td>2.36%</td>
<td>5.69%</td>
<td>76.43%</td>
</tr>
</tbody>
</table>

Table 5. RMSE and improvement with EKF at each axis of the online autonomous take-off and landing experiments

6. Conclusion

In this study, the ground stereo vision-based object aircraft detection and localization approach has been fused into flying object detection. An EKF estimator has been implemented to improve accuracy within scenarios, for instance, when the aircraft flies out of biocular views. By using the developed prototype of the ground stereo vision-based guidance system and the fixed-wing UAV, Pioneer, flying experiments have been conducted to collect sequential images and DGPS data. Both offline and online modes have been employed into performance evaluation of the spatial localization algorithms. The experimental results validate that the proposed approach effectively improves performances in localization accuracy, real-time capability and robustness.

7. Acknowledgements

This work was jointly supported by Major Application Basic Research Project of NUDT with Grant No. ZDYYJ-CY20140601 and the Scientific Research Foundation of Graduate School of NUDT. The authors would like to thank Boxin Zhao, Zhaowei Ma, Shulong Zhao, Zhiewei Zhong, Dou Hu and others for their contribution in building the ground stereo vision navigation system as our experimental platform.

8. References

on Intelligent Robots and Systems (IROS); 2013. p. 2971-2976


[26] Conway AR. Autonomous Control of an Unstable Helicopter Using Carrier Phase GPS only [thesis]. Stanford University; 1995


