

Ground Stereo-Vision-in-Loop Guidance for Safe Landing of Fixed-wing UAVs: Architecture & Algorithms

Weiwei Kong, Tianjiang Hu, Daibing Zhang, Lincheng Shen, Jianwei Zhang

Abstract

Orchestrating a safe landing is one of the greatest challenges for aircraft. This paper aims to deal with the problem of automatically landing a Unmanned Aircraft Vehicle (UAV) in the field with the help of a ground-based visual system. In this paper, A novel visual system consist of two separate perception modules and each has a camera mounting on a pan/tilt unit (PTU). These two modules setup at the each side of the runway and benefiting from the large baseline and extended field of view (FOV), the aircraft could be tracked at 1 *km* in distance. The position of the UAV could be calculated by triangular geometry. We provide the results from both simulation and practical experiments, showing the performance of the novel system and the overall accuracy during the landing process. Our approach offer increased accuracy in measuring aircraft position in GNSS-denied scenario when compare to human operation. This autonomous landing system caters for all of the different UAV system in operation, such as fixed-wing and rotary wing.

Keywords: UAV; Stereo Vision; Localization

1. Introduction

In the past decades, the use of unmanned aerial vehicles (UAVs) has increased tremendously in both military and civil fields. Although aerial robots have successfully implemented in several applications, there are still new research directions relate to them. Kumar et al. [1] summarized the

opportunities and challenges of this growing field, from the model design to high-level perception capability. All of these issues are concentrating on improving the degree of autonomy, which supports UAVs continue to be used in novel and surprising ways.

No matter fixed-wing platform or rotors aircraft, a standard fully unmanned autonomous system(UAS) involves performing takeoffs, waypoint navigation and landings. Among them, automatic landing is the most delicate and critical phase of a UAV flight. Two technical reports [2, 3], claimed that nearly 50% of fixed-wing UAVs such as Hunter and Pioneer operated by the US military suffer accidents during landing. Also, almost 70% of mishaps for Pioneer UAVs occurred during landing due to human factors. So a proper assist system is needed to enhance the reliability of landing task. Generally, two main capabilities of the system are required. The first one is navigation and localization of UAVs, and the second one is generating the appropriate guidance command to guide UAVs for a safe landing. In this paper, we mainly focus on the first issue and try to improve the guidance accuracy and robustness.

For manned aircraft, the traditional landing system uses a radio beam directed upward from the ground [4, 5]. By measuring the angular deviation from the beam through onboard equipment, the pilot knows the perpendicular displacement of the aircraft in the vertical channel. For the azimuth information, additional equipment is required. However, due to the size, weight, and pow-

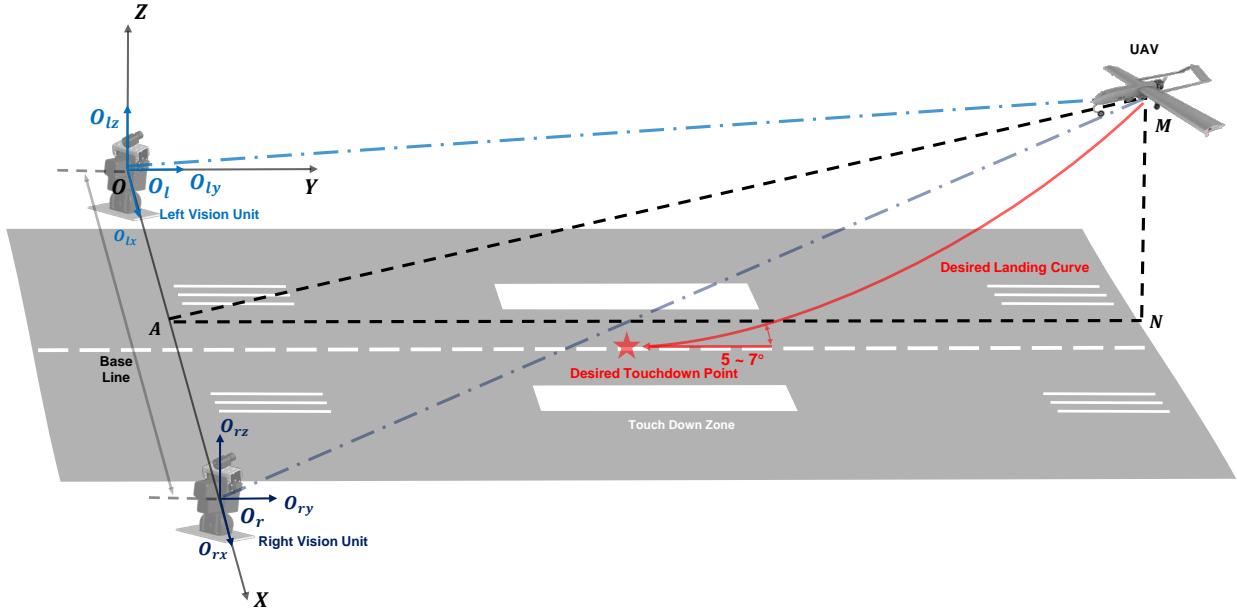


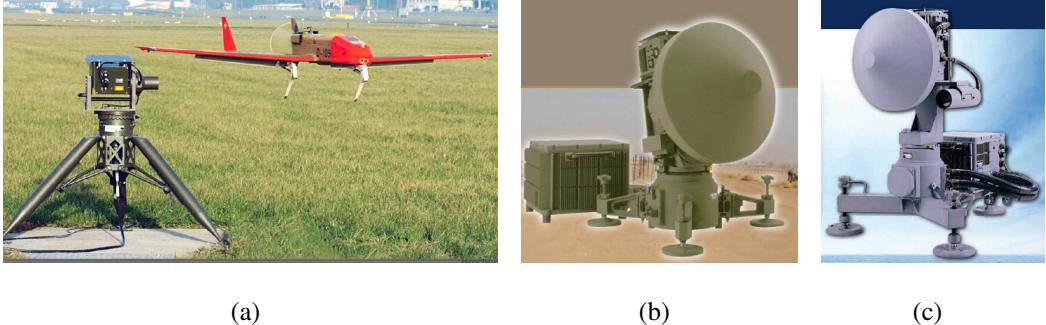
Figure 1: The geometry of one PTU with respect to optical center and image plane.

er (SWaP) constraints, it is impossible to equip these instruments in UAV. Thanks to the GNSS technology, we have seen lots of successful practical applications of autonomous UAVs in outdoor environments such as transportation, aerial photography and intelligent farming. Unfortunately, in some circumstances, such as urban or low altitude operations, the GNSS receiver antenna is prone to lose line-of-sight with satellites and making GNSS unable to deliver high quality position information [6]. So the autonomous landing in unknown or Global Navigation Satellite System(GNSS)-denied environment is still an open problem.

Visual based approach is an obvious way to achieve the autonomous landing by estimating flight speed and distance to the landing area, in a moment-to-moment fashion. Generally, two types of visual methods can be considered. The first type is the vision-based onboard system, which has been widely studied. The other is to guidance the aircraft using the ground-based camera system. Once the aircraft is detected by the camera during the landing process, its characteristics, such as type, location, heading and velocity, can be derived by the guidance system. Based on these information, the UAV could align itself carefully towards the landing area and adapt its velocity and acceleration to achieve safely landing. In summary, two key elements of the landing problem are detecting the UAV and its motion calculating the location of the UAV relative to the landing filed.

Motivated by these pressing problems, we introduce a novel ground-based visual landing system. Unless the existing ground-based guidance system that has limited baseline, our proposed approach sets up the camera on a pan-tilt unit (PTU) to increase the effective field of view (FOV) of cameras and the distance at which a reliable lock could be established and maintained. The general view of the fresh guidance system is shown in Figure 1. The key contributions of this papers is threefold:

- First, we introduce a novel ground-based guidance system with large baseline and 360 circular coverage, eliminating the dependency of GNSS.
- Second, state-of-the-art image processing algorithms, Active Contour and AdaBoost and modified TLD framework have been applied to the problem of target detection and tracking under different light condition in the real time experiments.
- Third, the whole system has been tested using a fixed-wing aircraft with an appropriate



(a)

(b)

(c)

Figure 2: (a) OPATS (b) DT-ALTS (c) UCARS-V2

landing strategy.

2. Related Works

2.1. Instrument Approach

To achieve better performance in GNSS-denied environment, some military-grade on-ground systems, such as OPATS [7], DT-ATLS [8] and UCARS [9], were developed, mainly based on millimeter wave track radar, laser pointer, cooled infrared camera, etc. Yet no further information about these products is publicly available. These systems are shown in Figure.2(a), Figure.2(b) and Figure.2(c).

The Instrument Landing System (ILS) is widely used in most of the international airports around the world allowing pilots to establish on the approach and follow the ILS, in autopilot or not, until the decision height is reached. This approach refer to a ground-based instrument system providing accurate guidance to a plane takeing-off from a runway or approaching and landing on a runway. A reference book [10] provided in-depth discussion of the ILS.

2.2. On-board Vision Method

The problem of accurately landing a UAV using vision-based control has been well developed for autonomous rotorcraft. Visual Servo algorithms have been extensively studied in the robotics filed over the past decade. Considering the model as an eye-in-hind configuration, an practical imaged-based visual servo control frame work has been proposed by Guenard [11]. By calculating the image error kinematics, a designed Lyapunov controller shows good performance and robustness

of hovering a quadrotor UAV.

Mejías developed an image-based velocity references approach to visually control the altitude and lateral of the helicopter [12]. These vision velocity reference is combined with GPS and IMU data for more accurate and position measurements.

An alternative to template-based method are optical-flow-aided position measurement systems. This approach was inspired by flying insect navigation strategies [13] and have been developed over the years to use onboard terrestrial and aerial robots. Triggered by the observations of honeybees, an optic flow regulator was introduced [14]. It allows a tethered rotorcraft to deal with non-stationary environments and the rotorcraft was able to land safely near a target which mounted on a moving platform both vertically and horizontally. In another recent work [15], a discrete-time non-linear model predictive controller (MPC) was employed to land a Parrot AR.Drone on a Pioneer mobile robot with inclined platform. The on-board forward facing camera was used for tracking the moving target and the downwards pointing camera for optical flow. For fixed-wing aircraft, Kim et al. [16] reported a vision-based net recovery system. The ground station received the image capture by the on-board camera aiming to detect a recovery net. The longitudinal and lateral bearing angle were calculated in keeping with a guidance law based on the net relative position. This technique is not suitable for the whole landing process because the vision operating range is only 50 m.

For cooperative localization some structured landing mark, such as 2D barcodes, helipad, could provide the relative transform between landing area and camera. Sharp et al. [17] solved the landing task of a Yamaha R-50 helicopter by using the designed landing target(white squares) for landing area recognition and estimating the relative position. By calculating the multiple view matrix, their proposed algorithms [18] improve the motion and structure estimation. However, the controller only regulates the x and y position to hover over the landing platform. Saripalli et al. also presented a image-moment-based method for autonomous landing of a helicopter on a H-shape landing pad [19]. However, the precise height estimation benefits from the differential GPS system. In 2012, a worthwhile study [20] provide a framework for an automated vision-based recovery of an unmanned rotatory craft onto a moving platform. The onboard camera system calculates the displacement and orientation of the UAV respect to the pattern and autonomous maneuvers the craft towards the landing platform. The workable limit for robust tracking was 10

m. For fixed-wing landing, German Aerospace Center (DLR) built and tested the 44-pound drone which successfully spotted a QR code on the roof of an Audi and touched down on the net affixed to the station wagons roof, at 43 mph [21]. However, due to the viewing angle of the lens and the size of fiducial marker, the detection distance of the sole of vision control method is limited.

The combination of laser and visual system also has a long history in this field. In 2009, Garratt [22] considered a relatively low-cost guidance system through visual tracking and LIDAR positioning for autonomous recovery of a Yamaha R-max rotorcraft from ships at sea. This on-board visual system identifies and tracks the single beacon to determine both the distance and the orientation of the deck. Because the wavelength of the beacon is 650 nm, the craft can track it in bright sunlight. For the laser system, the laser rangefinder at a wavelength of 780 nm assembling with a spinning mirror was mounted at the bottom of the rotorcraft, which scans a conical pattern on the deck for estimating the position. The error of position estimation is better than 2 cm in practice, but the detection distance is only 100 m due to the size and the power of the selected LED.

While there is significant work using electro-optical pod, laser scanner and radar, those methods require more hardware for the aircraft to carry; therefore, in this case, we focus just on ground-based visual case, which is a lighter weight solution for UAV.

2.3. Ground-based Vision Method

While several techniques have been applied for on-board vision-based control of UAVs, few have shown landing of a fixed-wing guiding by ground-based vision system. Wang [23] proposed a system using a step motor controlling web camera to track and guide a micro-aircraft. The drawback of this system is the limited baseline leading to a narrow field of view (FOV). To increase the camera FOV, multi-camera systems are considered attractive. This kind of system could solve the common vision problems and track object to compute their 3D locations. At Chiba University [24], a ground-based Bumblebee stereo vision system was used to calculate the 3D position of a quadrotor at the altitude of 6 meter.

In addition, Martinez [25] introduced a trinocular on-ground system, which is composed by three or more cameras for extracting key features of UAV in order to obtain robust 3D position estimation. The range of detection is still not sufficient for fixed-wing UAV. And another draw-

back of multi-camera system is the calibration process, whose parameters are nontrivial to obtain. Moreover, we also reviewed various vision-based landing approaches performing on different platform [26] and Gautam etc. provide another general review of the autonomous landing techniques for UAVs [27].

3. Ground Stereo System Architecture

In this section, we introduce the theoretical model for ground-to-air visual system. We first recap the traditional stereo vision model which has limited base line restraining the detection distance. To enlarge the system working boundary, we setup the camera and other sensor module on the two separated PTU and then calculate the target according the image information and rotation angle from PTU.

3.1. Traditional Stereo Vision Model

The standard camera model is pin-hole camera Mode. The coordinate of the target M is (x, y, z) , and its position on image plane is (u, v) . The camera focus is f , then the relationship of coordinate between the 3D world and 2D image plane is

$$\lambda \begin{bmatrix} u \\ v \\ f \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (1)$$

where λ is the scale factor. When using euclidean coordinate, we could get

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix} \quad (2)$$

Assuming the camera focus f and the image size (w_u, w_v) is known, the horizontal FOV of camera α_{FOV} can be calculated by

$$\tan \frac{\alpha_{FOV}}{2} = \frac{w_u}{f} \quad (3)$$

Similarly, we can prove that the vertical FOV of camera β_{FOV} is

$$\tan \frac{\beta_{FOV}}{2} = \frac{w_v}{f} \quad (4)$$

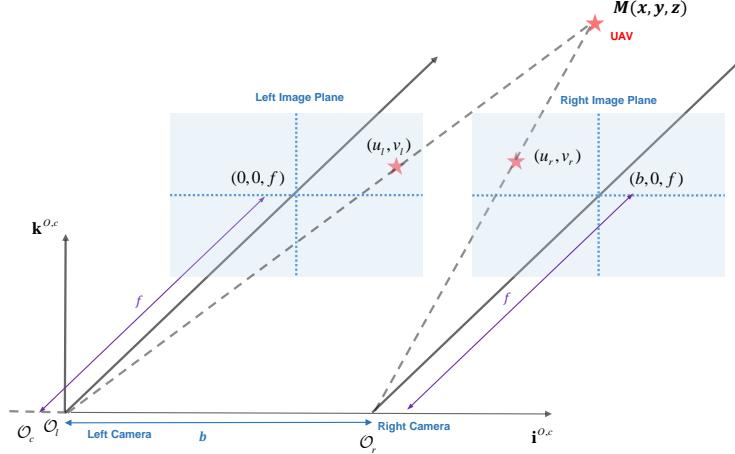


Figure 3: Theoretical Stereo Vision Model

When calculating the FOV, the unit of f and (w_u, w_v) should be same, such as pixel or mm.

Although the above model is simple, it could be helpful to estimate the theoretical camera lens according the expected distance and resolution or to measure the target size roughly based on the pixel length on image plane. Let the focus length be f , the width and height of the target be W and H , the distance between the camera and target be L , the target projection on image plane be w and h , the relationship between them is

$$f = \frac{wL}{W} \quad (5)$$

$$f = \frac{hL}{H} \quad (6)$$

When the optical axes of these two cameras are parallel, we could calculate the target in 3D space by

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \frac{b}{d} \begin{bmatrix} u_l \\ u_r \\ f \end{bmatrix} \quad (7)$$

where b is the baseline and $d = u_l - u_r$ is the pixel disparity. The traditional stereo vision model as shown in Fig. 3. Even though some calibration methods could manage the axes unparallel situation, it is still difficult to calculate the system correctly as the baseline is large.

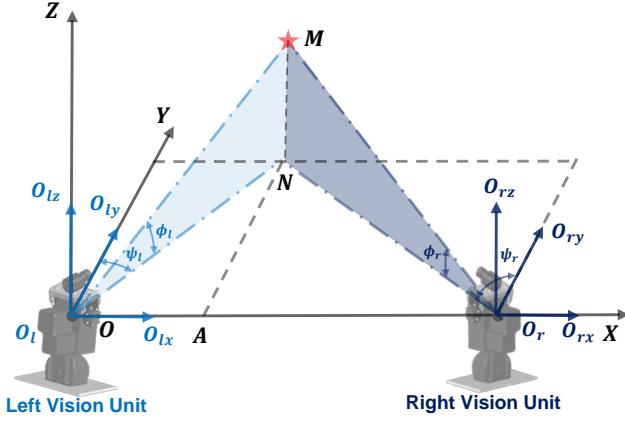


Figure 4: Theoretical Optical Model. The target M is projected at the center of the optical axis.

3.2. Separate Stereo Vision System Configuration

In order to detect the target in long distance, the large base line, more than 5 m is required. Benefiting the camera assembled on PTU separately, we could switch the baseline freely according to the expected detection distance and target size.

In this paper, we assumed that the world coordinate system (X, Y, Z) is located on the origin of the left vision unit, the rotation center of the PTU. For the sake of simplicity, the camera is installed on the PTU in the way that the axes of the camera frame are parallel to those of PTU frame. The origins of these two frames are close. So it can be assumed that the camera frame coincides with the body frame. Fig. 4 reveals the theoretical model for visual measurement. After installing the right camera system on X -axis, the left and right optical center can be expressed as O_l and O_r respectively. Then the base line of the optical system is O_lO_r whose distance is D . Considering the center of mass of UAV as a point M , O_lM and O_rM illustrate the connections between the each optical center and the UAV. In addition, ϕ_l , ϕ_r , ψ_l , ψ_r denote the tilt and pan angle on both side. Therefore, we define $\phi_l = 0$, $\phi_r = 0$, $\psi_l = 0$ and $\psi_r = 0$, as the PTU is set to the initial state, i.e. the optical axis parallel the runway; the measurement of a counterclockwise direction is positive.

Since the point M not coincides with the principle point which is the center of image plane, the pixel deviation compensation in the longitudinal and horizontal direction should be considered. As shown in Fig. 5, we calculate pixel deviation compensation on the left side by

$$\begin{cases} \psi_{cl} = \arctan \frac{(u - u_0)du}{f} \\ \phi_{cl} = \arctan \frac{(v - v_0) \cos \psi_{cl} dv}{f} \end{cases} \quad (8)$$

where optical point is $o(u_o, v_o)$, du and dv are the pixel length of the u - and v -axis in image plane, and f is the focus. The current PTU rotation angle can be directly obtained through the serial ports during the experiments. Let ϕ_{pl} and ψ_{pl} be the left pan and tilt angle separately. Then, the total pan and tilt angle on the left side can be detailed as:

$$\begin{cases} \phi_l = \phi_{cl} + \phi_{pl} \\ \psi_l = \psi_{cr} + \psi_{pr} \end{cases} \quad (9)$$

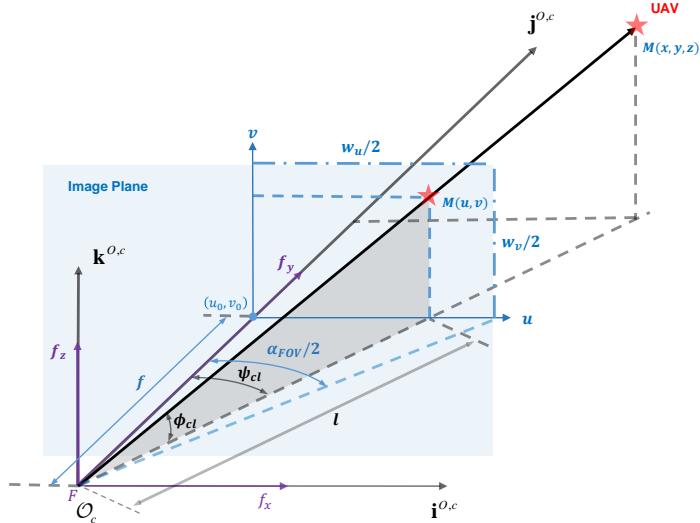


Figure 5: The geometry of one PTU with respect to optical center and image plane.

For the other side, we could also calculate the angle in the same way.

The world coordinates of point M is $(x_M, y_M, z_M) \in \mathbf{R}^3$. Point N is the vertical projection of point M on XOY plane, and NA is perpendicular to X -axis. If we define $NA = h$, the following guidance parameters can be obtained:

$$\begin{cases} x_M = h \tan \psi_l = \frac{D \tan \psi_l}{\tan \psi_l - \tan \psi_r} \\ y_M = h = \frac{D}{\tan \psi_l - \tan \psi_r} \\ z_M = \frac{h \tan \phi_l}{\cos \psi_l} = \frac{D \tan \phi_l}{\cos \psi_l (\tan \psi_l - \tan \psi_r)} \end{cases} \quad (10)$$

Also, errors in the internal and external camera calibration parameters marginally affects some of the estimates - the x-position and z-position, in particular.

3.2.1. Error Analysis

We are now in the position to analysis the error related to the PTU rotation angle. According to (10), the partial derivative of each equation respect to the pan angle and the tilt angle are denoted in the following way,

$$\begin{cases} \frac{\partial x_M}{\partial \psi_l} = \frac{D \tan \psi_r}{\cos^2 \psi_l (\tan \psi_l - \tan \psi_r)^2} \\ \frac{\partial x_M}{\partial \psi_r} = \frac{D \tan \psi_l}{\cos^2 \psi_r (\tan \psi_l - \tan \psi_r)^2} \end{cases} \quad (11)$$

$$\begin{cases} \frac{\partial y_M}{\partial \psi_l} = \frac{D}{\cos^2 \psi_l (\tan \psi_l - \tan \psi_r)^2} \\ \frac{\partial y_M}{\partial \psi_r} = \frac{D}{\cos^2 \psi_r (\tan \psi_l - \tan \psi_r)^2} \end{cases} \quad (12)$$

$$\begin{cases} \frac{\partial z_M}{\partial \phi_l} = \frac{D}{\cos \psi_l \cos^2 \phi_l (\tan \psi_l - \tan \psi_r)} \\ \frac{\partial z_M}{\partial \psi_l} = \frac{D \tan \phi_l (\cos \psi_l + \sin \psi_l \tan \psi_r)}{\cos^2 \psi_r (\tan \psi_l - \tan \psi_r)^2} \\ \frac{\partial z_M}{\partial \psi_r} = \frac{D \tan \phi_l}{\cos \psi_l \cos^2 \psi_r (\tan \psi_l - \tan \psi_r)^2} \end{cases} \quad (13)$$

To analyze the influence of the error from the angle, we define the gradient of the world coordinate as

$$\nabla_{x_M}(\psi_l, \psi_r) := \left(\frac{\partial x_M}{\partial \psi_l}(\psi_l, \psi_r), \frac{\partial x_M}{\partial \psi_r}(\psi_l, \psi_r) \right) \quad (14)$$

$$\nabla_{y_M}(\psi_l, \psi_r) := \left(\frac{\partial y_M}{\partial \psi_l}(\psi_l, \psi_r), \frac{\partial y_M}{\partial \psi_r}(\psi_l, \psi_r) \right) \quad (15)$$

$$\nabla_{z_M}(\psi_l, \psi_r) := \left(\frac{\partial z_M}{\partial \psi_l}(\psi_l, \psi_r), \frac{\partial z_M}{\partial \psi_r}(\psi_l, \psi_r) \right) \quad (16)$$

In this case, simulation is needed to evaluate the behavior of our visual system. Fig. 6(a), Fig. 6(b), and Fig. 6(c) are the vector field distribution of $\nabla_{x_M}(\psi_l, \psi_r)$, $\nabla_{y_M}(\psi_l, \psi_r)$, and $\nabla_{z_M}(\psi_l, \psi_r)$, which give us an intuitive result under different types of errors. The length of each vector describes the strength at a specific point; the direction along the vector points to the direction of the fastest

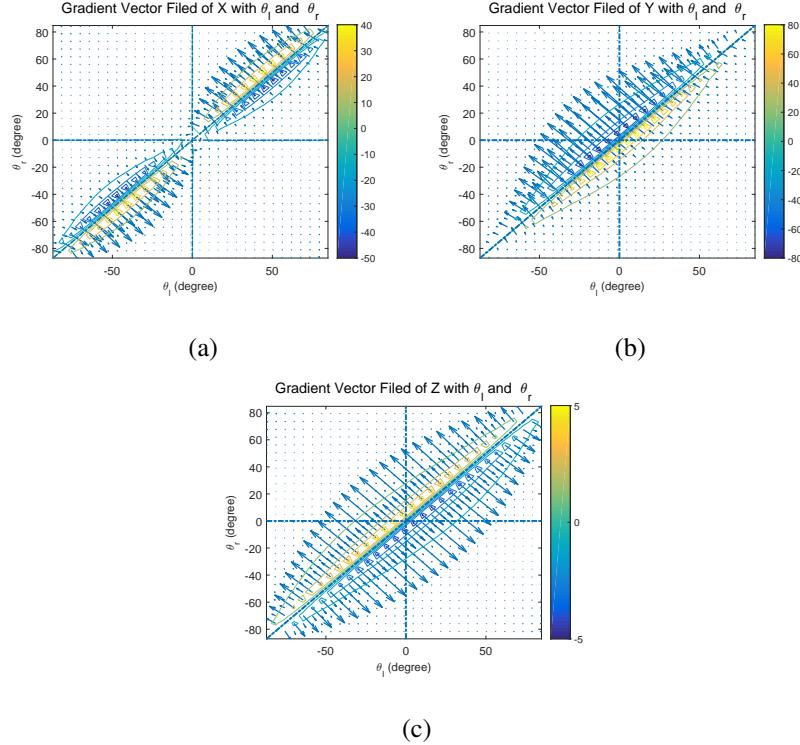


Figure 6: Vector Field Distribution of $\nabla_{x_M}(\psi_l, \psi_r)$, $\nabla_{y_M}(\psi_l, \psi_r)$, and $\nabla_{z_M}(\psi_l, \psi_r)$

error increase. However, only when $y_M \geq 0$ (the aircraft is in front of two cameras), the area $\psi_l - \psi_r > 0$ has the physics meaning. Fig. 7(a) shows that x_M has a significant variation when ψ_l is approximate to ψ_r , namely the optical axes are nearly parallel. Further, y_M and z_M have the similar variations. Considering the general working status of the ground-based system, we mainly focus on the second quadrant of aforementioned vector fields as shown in 7(a), 7(b) and 7(c). In these area, there are slight variation that theoretically demonstrates the feasibility of the system.

3.2.2. More Practical Analysis

In theory, O_lM and O_rM should intersect perfectly at one point all the time as shown in Fig. 4. Due to the inevitable errors from PTU rotation and tracking algorithms, we estimates the intersecting point by combing the vertical line of two different plane in space.

(1) We set $(x_{ol}, y_{ol}, z_{ol}) = (0, 0, 0)$ and $(x_{or}, y_{or}, z_{or}) = (D, 0, 0)$ are the optical center of each camera. Assuming that $a_l \neq 0, b_l \neq 0, c_l \neq 0$ and $a_r \neq 0, b_r \neq 0, c_r \neq 0$, we obtain the parametric

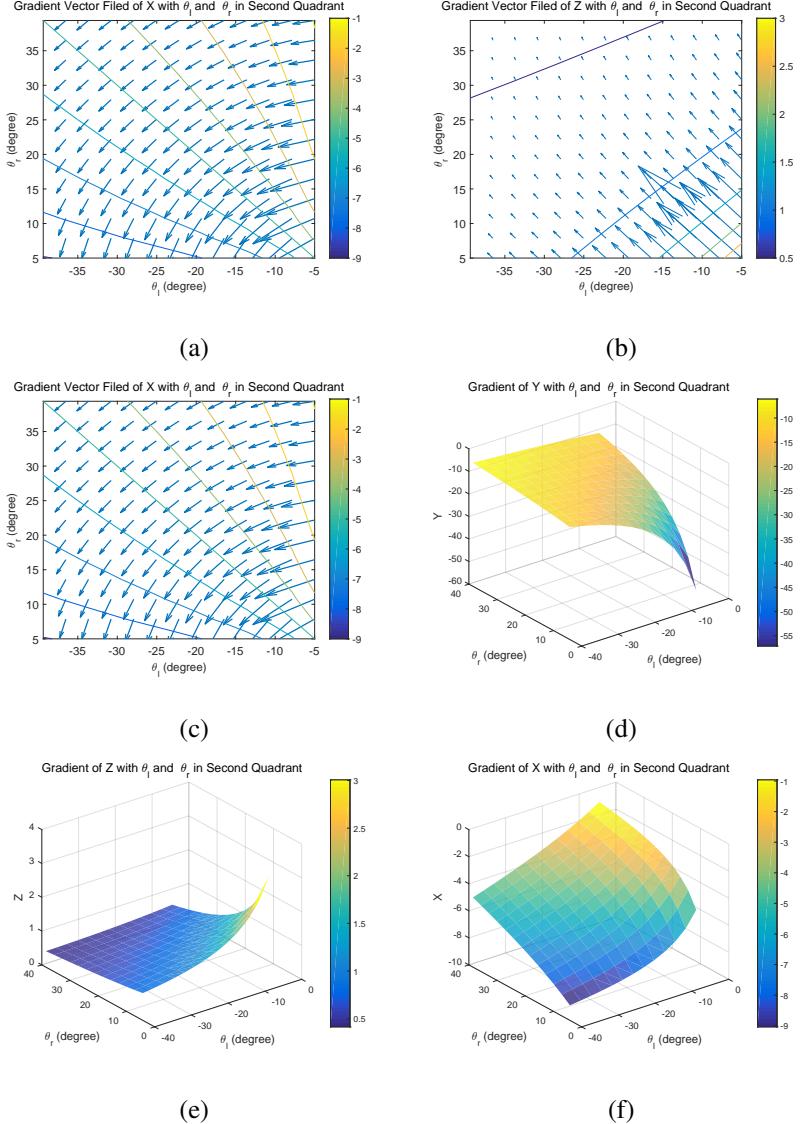


Figure 7: (a)-(c) Vector Field Distribution of $\nabla_{x_M}(\psi_l, \psi_r)$, $\nabla_{y_M}(\psi_l, \psi_r)$, and $\nabla_{z_M}(\psi_l, \psi_r)$ in Second Quadrant. (d)-(f) Gradient of X, Y, Z with θ_l and θ_r in Second Quadrant

equations of line O_lM and O_rM

$$\begin{cases} \frac{x - x_{ol}}{a_l} = \frac{y - y_{ol}}{b_l} = \frac{z - z_{ol}}{c_l} = t_l, \\ \frac{x - x_{or}}{a_r} = \frac{y - y_{or}}{b_r} = \frac{z - z_{or}}{c_r} = t_r, \end{cases} \quad (17)$$

$$\begin{cases} a_l = \cos \phi_l \sin \psi_l \\ b_l = \cos \phi_l \cos \psi_l \\ c_l = \sin \phi_l \end{cases} \quad (18)$$

$$\begin{cases} a_r = \cos \phi_r \sin \psi_r \\ b_r = \cos \phi_r \cos \psi_r \\ c_r = \sin \phi_r \end{cases} \quad (19)$$

where t_l, t_r are the parameters for the line O_lM and O_rM separately. So any point (x, y, z) on each line are usually written parametrically as a function of t_l and t_r :

$$\begin{cases} x_l = a_l t_l + x_{ol} \\ y_l = b_l t_l + y_{ol} \\ z_l = c_l t_l + z_{ol} \end{cases} \quad (20)$$

$$\begin{cases} x_r = a_r t_r + x_{or} \\ y_r = b_r t_r + y_{or} \\ z_r = c_r t_r + z_{or} \end{cases} \quad (21)$$

(2) In our situation, O_lM and O_rM are skew lines that these two lines to not be parallel and to not intersect in 3D. Generally, the shortest distance between the two skew lines lies along the line which is perpendicular to both of them. By defining the intersection points of the shortest segment line for each line by (x_{lp}, y_{lp}, z_{lp}) and (x_{rp}, y_{rp}, z_{rp}) , we get the parametric equations

$$\begin{cases} x_{lp} = a_l t_l + x_{ol} \\ y_{lp} = b_l t_l + y_{ol} \\ z_{lp} = c_l t_l + z_{ol} \end{cases} \quad (22)$$

$$\begin{cases} x_{rp} = a_r t_r + x_{or} \\ y_{rp} = b_r t_r + y_{or} \\ z_{rp} = c_r t_r + z_{or} \end{cases} \quad (23)$$

(3) Knowing the position of the intersection points on each line, the distance is calculated by the square Euclidean norm

$$J = \|(x_{lp}, y_{lp}, z_{lp}) - (x_{rp}, y_{rp}, z_{rp})\|_2^2 \quad (24)$$

The purpose of this method is to minimize this distance. Since two skew lines can always be placed into two parallel planes, we could simplify the problem to finding the shortest distance between

the two parallel planes. By taking the partial derivative of the distance function J with respect to t_l and t_r and setting them to 0, the above function can be calculated as:

$$\begin{aligned} & \begin{bmatrix} a_l^2 + b_l^2 + c_l^2 & -(a_l a_r + b_l b_r + c_l c_r) \\ -(a_l a_r + b_l b_r + c_l c_r) & a_l^2 + b_l^2 + c_l^2 \end{bmatrix} \begin{bmatrix} t_l \\ t_r \end{bmatrix} \\ &= (x_{ol} - x_{or}) \begin{bmatrix} -a_l \\ a_r \end{bmatrix} + (y_{ol} - y_{or}) \begin{bmatrix} -b_l \\ b_r \end{bmatrix} \\ &+ (z_{ol} - z_{or}) \begin{bmatrix} -c_l \\ c_r \end{bmatrix}. \end{aligned} \quad (25)$$

We could then define the matrix as follows:

$$\mathbf{H} = \begin{bmatrix} a_l^2 + b_l^2 + c_l^2 & -(a_l a_r + b_l b_r + c_l c_r) \\ -(a_l a_r + b_l b_r + c_l c_r) & a_l^2 + b_l^2 + c_l^2 \end{bmatrix} \quad (26)$$

The determinant of this matrix is denoted $\det \mathbf{H}$. When $\det \mathbf{H} = 0$, MO_l and MO_r parallel to each other; when $\det \mathbf{H} \neq 0$, there is an uniqueness vertical line. After solving these equations, the parameter t_l and t_r can be expressed as

$$\begin{cases} t_l = D \frac{a_l(a_l^2 + b_l^2 + c_l^2) - a_r(a_l a_r + b_l b_r + c_l c_r)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} \\ t_r = D \frac{a_l(a_l a_r + b_l b_r + c_l c_r) - a_r(a_l^2 + b_l^2 + c_l^2)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} \end{cases} \quad (27)$$

and the two intersecting points of public vertical line are

$$\begin{cases} x_{lp} = a_l D \frac{a_l(a_l^2 + b_l^2 + c_l^2) - a_r(a_l a_r + b_l b_r + c_l c_r)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} \\ y_{lp} = b_l D \frac{a_l(a_l^2 + b_l^2 + c_l^2) - a_r(a_l a_r + b_l b_r + c_l c_r)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} \\ z_{lp} = c_l D \frac{a_l(a_l^2 + b_l^2 + c_l^2) - a_r(a_l a_r + b_l b_r + c_l c_r)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} \end{cases} \quad (28)$$

and

$$\begin{cases} x_{rp} = D \left[a_r \frac{a_l(a_l a_r + b_l b_r + c_l c_r) - a_r(a_l^2 + b_l^2 + c_l^2)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} + 1 \right] \\ y_{rp} = b_r D \frac{a_l(a_l a_r + b_l b_r + c_l c_r) - a_r(a_l^2 + b_l^2 + c_l^2)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} \\ z_{rp} = c_r D \frac{a_l(a_l a_r + b_l b_r + c_l c_r) - a_r(a_l^2 + b_l^2 + c_l^2)}{(a_l b_r - b_l a_r)^2 + (b_l c_r - c_l b_r)^2 + (a_l c_r - c_l a_r)^2} \end{cases} \quad (29)$$

Further, the optimal point to present point M is the center point of the vertical line segment, which is satisfied the least square method. To reach a more general result, we added weight on the coordinates of each intersecting points and the world coordinate of the target point M is

$$\begin{bmatrix} x_M \\ y_M \\ z_M \end{bmatrix} = w \begin{bmatrix} x_{lp} \\ y_{lp} \\ z_{lp} \end{bmatrix} + (1 - w) \begin{bmatrix} x_{rp} \\ y_{rp} \\ z_{rp} \end{bmatrix}, w \in [0, 1]. \quad (30)$$

The angle between the UAV landing trajectory and the runway area is usually between 5° and 7° . By considering 1 mrad normal distributed disturbance (the accuracy of the PTU is 0.006°), Fig. 8 illustrates measurement errors of x_M , y_M and z_M in case of different points $(x, y) \in S$, where $S = \{(x, y) | -50 \leq x \leq 50, 20 \leq y \leq 1000\}$. Obviously, the errors at a considerable distance are notable, but the incidence of them decline while the aircraft is close to the runway. When the UAV is only 100 m to the landing area, the error of altitude is about 0.02 m that is dependable for the landing task. The further evaluations in field environment are discussed in Section V-B.

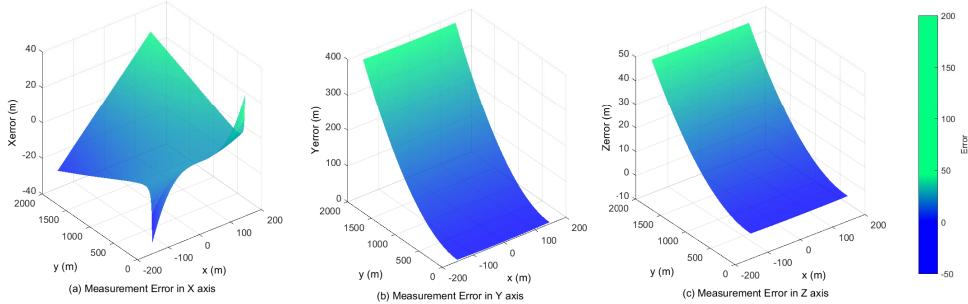


Figure 8: Baseline is 15 m, Focus is 100 mm, The Mersurement Errors in X , Z and Y axis.

3.3. Implementation

For visible light camera, we selected DFK 23G445 which developed by Imaging Source GmbH shown in Fig. 10(a). The sensor of this camera is Sony ICX445AQA equipped with GigE interface which has fast data transfer rates, typically up to 1000 Mbit/s. This camera has an image resolution of 1280×960 with RGB32 color model, a maximum frame rate of 30 fps. The lens of the vision system we adopted is 100 mm, and the baseline is 10 m. The ground station, which runs the

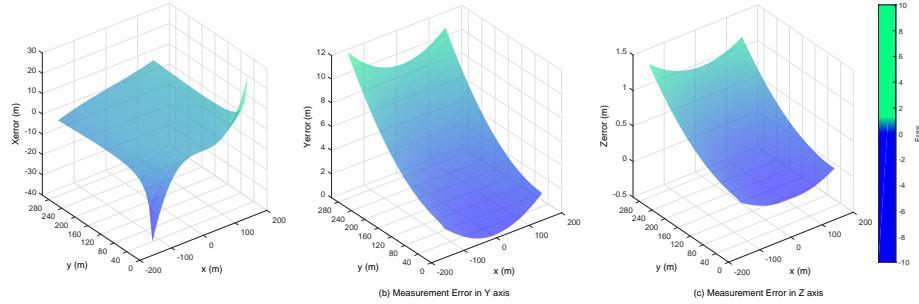


Figure 9: The geometry of one PTU with respect to optical center and image plane.

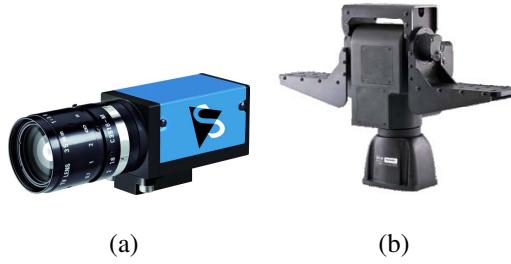


Figure 10: (a) DFK 23G445 Camera (b) PTU-D300E

tracking and localization algorithms, is mainly constituted by ADLINK EOS-1200 PC, which sends relative position and records GNSS data from UAV as ground truth. This product is a rugged and compact system equipped with Intel Core i7 2.1 GHz processor and 8 GB DDR3.

To extend the field of view, we adopted precision PTU to actuate the camera. PTU-D300E (see Fig. 10(b)) is a high performance product from FLIR. Its pan/tilt speeds up to $50^\circ/\text{second}$ with the position resolution of 0.006° . Moreover, it is a user programmable product integrating Ethernet and RS-232 interface. The real-time command interface supports advanced applications such as video tracking. We set up the camera on the top bracketing, and the assembled individual vision system is illustrated in Fig. 12(a).

For short-baseline configuration, cameras were setup on one PTU (Fig. 11) and the system should mount on the center line of the runway. However, this system has limited range for UAV detection. According to our discussion in last section, we could also fix the cameras with separated PTU on the both sides of the runway, as shown in Fig. 12(b), so the landing aircraft can be locked by our system around 1 km.

The overall architecture of the separated stereo system and its connections and data flows are shown in Fig. 13. Each vision unit work independently and transfers the results of image process-



Figure 11: Short-baseline Ground System

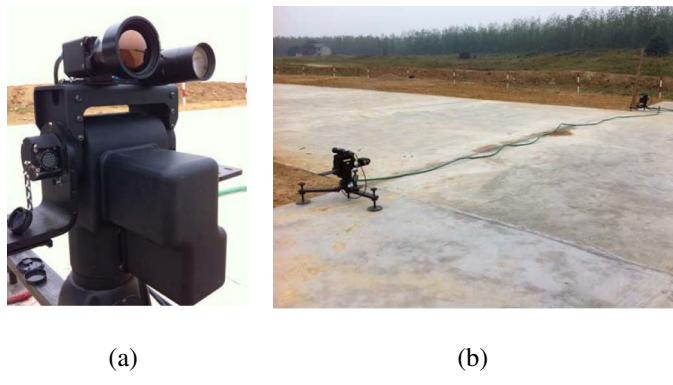


Figure 12: (a) Assembled vision system. (b) The setup of the visual system.

ing and PTU status to the navigation computer which calculates the estimated relative position of the UAV. Since the ground station judging the switch from GNSS-aid method to vision-aid method, the auto pilot guided UAV based on the properly data received by XTend modem.

4. Localization-related Algorithms

To keep the camera on the PTU pointing to the target closely, we designed a vision tracking framework based on TLD and active contour algorithm. Also, the camera rotation angle in pan and tilt axis, which can be received through the serial port from PTU, can also improve the tracking accuracy. Fig. 14 shows the tracking framework.

4.1. SRC-TLD Algorithm

TLD is an adaptive tracking-by-learning approach, which tracks by learning the appearance of the target - using the online learned classifier to detect the object during the progress. It is consist of

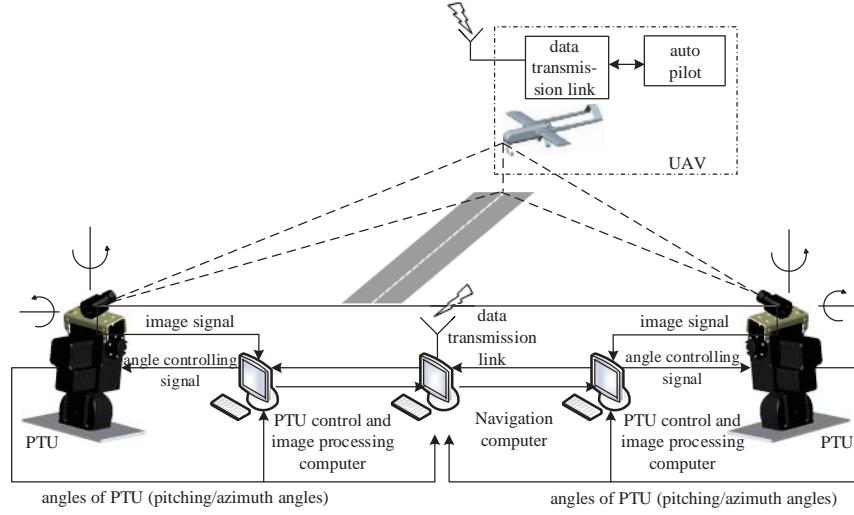


Figure 13: Architecture of the system.

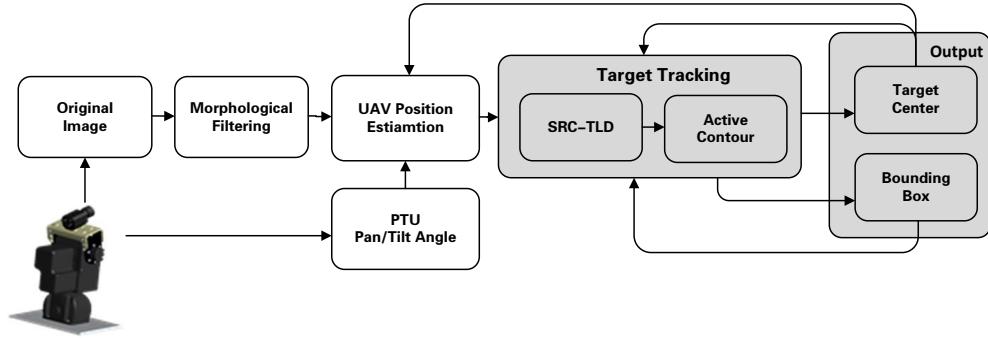


Figure 14: PTU-based UAV Tracking Framework

three components:

- Tracker is based on Lucas-Kanade method, which could track the target frame by frame and generate examples for learning components.
- Detector is to solve the classification problem based on random forest algorithm.
- Learning uses trackers to generate positive and negative samples which is mainly based on P-N Learning mechanisms. Also it update the inner model of the detector.

All of these three components runs independently during the calculation.

To describe the target, we compare the intensity values of several pixels. An example of five-pair feature calculation for a single ferns shows in Fig.15. The random pairs method selects the five

different location randomly and the result of each of these comparisons is interpreted as a binary digit, 10011. When written in decimal form, we get 19. A probability could be obtained combining the other training data set.

However, the random pairs is suitable for the target, which could fully filled the subwindow, such as in face or pedestrian detection scenes. This kind of randomness could retrieve the feature of the target and adapts the variance during the tracking process. For fixed-wing MAV, this type of target only displays in the narrow part of the subwindow as shown in light yellow rectangle. Especially the wings of the MAV seems to be a single line when it is far away from the runway. The general randomness lose the kernel feature of the target occasionally, so we set one pixel of each pair randomly only in the yellow rectangle and the other pixel located in the rest part of the subwindow. We named this process Semi-random Coding (SRC).

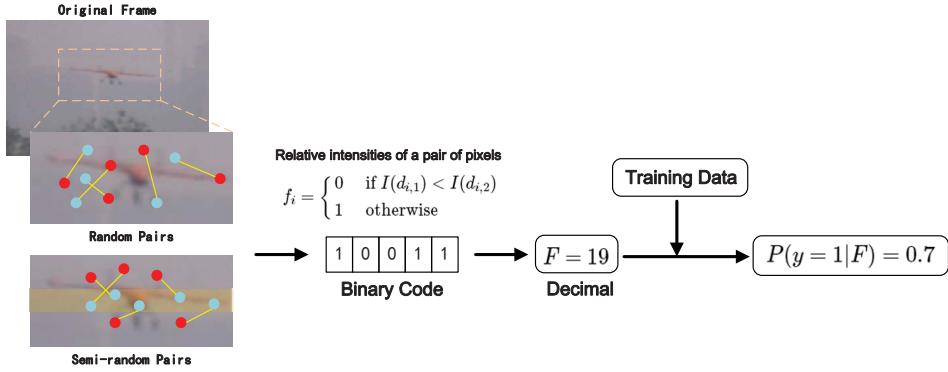


Figure 15: We compare the intensities of different pairs for each image to get the binary code, then store it as decimal. Random pairs are generated randomly through the whole image plane. For semi-random pairs, one of the point is located randomly in the light yellow led and the other point is generated randomly at the rest part.

4.2. Active Contour Algorithm

In order to enhance the localization accuracy, we need to detect the center of aircraft more precisely. However, the target center of SRC-TLD is the centroid of the bounding box which is not always true when the UAV was turning during the landing process. The left image of Fig. 16 shows the standard results of SRC-TLD results. The green bouding box is the ground truth, and the red one

is achieved from the SRC-TLD method. Active contour algorithm is one of the choice to converge the predefined rectangle to the UAV main body as shown in the right image of Fig. 16.

Some previous works [28] and [29], demonstrated that active contours are autonomous processes which employ image coherence in order to track various features of interest over time. A book [30] and the references therein described snakes fit very spontaneously into a control framework. The starting point of Level Set Method is [31], [32] in which an active contour model founded on the level set formulation of the Euclidean curve shortening equation is proposed. Specifically, the model is:

$$\frac{\partial \Psi}{\partial t} = \phi(x, y) \|\nabla \Psi\| (\operatorname{div}\left(\frac{\nabla \Psi}{\|\nabla \Psi\|}\right) + \nu) \quad (31)$$

the function $\phi(x, y)$ depends on the given image and is used as a “stopping term”. Generally the term $\phi(x, y)$ is selected to be small near an intensity-based edge and acts to stop evolution when the contour gets close to an edge.

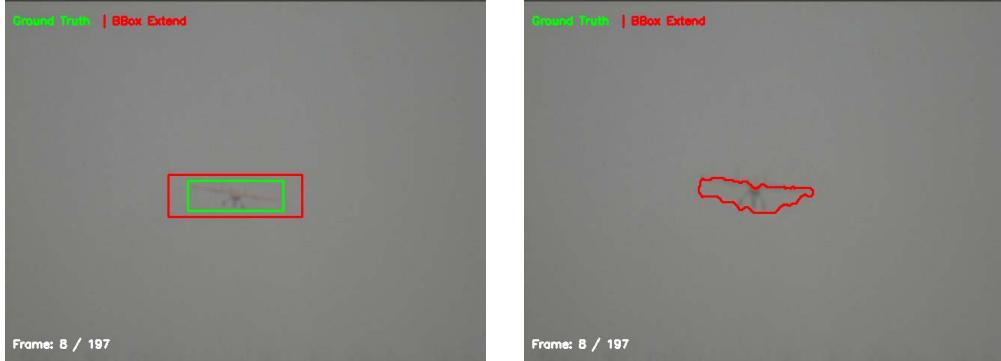


Figure 16: Active Contour Algorithm

5. Experiments and Discussion

5.1. UAV Platform

In our on-board system we can see three components: The autopilot design allowed for the aircraft to perform simple commanded maneuver. The aircraft was flying at 60 m altitude at 25 m/s. This experimental test-bed is also a customized fixed wing aircraft as shown in Fig. 17. It is a gasoline-powered radio-controlled model aircraft approximately 2.3 m in length and capable of lifting approximately 5 kg of payload. Table 1 lists the other technical specifications.



Figure 17: The Pioneer fixed-wing platform

Table 1: The Technical Specifications of Pioneer

Items	Description
Vehicle mass	9000 g
Maximum Payload mass	5000 g
Diameter	2900 mm
Flight duration	up to 180 minutes
Cruising speed	30.0 m/s

The autopilot module is iFLY-F1A (Fig. 18(a)). It consists of the F1A autopilot system, a ground control station, a redundant power management module and an engine RPM monitoring module. We further make use of the iFLY-G2 (Fig. 18(b)) navigation autopilot module [33], which is a small six-DOF (degree of freedom) navigation system. The G2 module includes a triaxial gyro, triaxial accelerometer, triaxial magnetometer, GNSS module, barometric altimeter, airspeed gauge and thermometer. It supports real-time 3D information including attitude angle, angular rate, position, speed, acceleration, true air speed, calibrated air speed. F1A is connected with G2 through RS-232 serial port.

Communication is crucial in the landing guidance framework, because the relative localization is broadcast through the radio. The guidance data are sent using an advanced radio modem that transmits and receives on the 900 Mhz band. The XTend RF Modems support up to 22 km outdoor communication with the interface data rates from 10 bps to 230,00 bps, which is sufficient to transfer GNSS data and predicted position from ground station to on-board navigation modem.

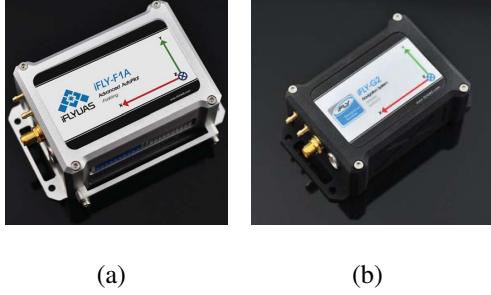


Figure 18: (a) iFLY-F1A Module (b) iFLY-G2 Module

5.2. SRC-TLD Experiments

To compare with the standard TLD method, the stability of Semi-random Code method was estimated for each frame by means of running 1000 Monte-Carlo experiments in one landing sequence. Tabel 2 lists the parameters for TLD and SRC-TLD and the result is shown as 19. The line shows the center location error for each algorithm, and the continuous shaded area is the error region, which explicate that the SRC has better stability and the accuracy of center location is improved.

Table 2: Parameters for TLD and SRC-TLD

Parameters	
Min/Max Scale	10
Features Number	10
Trees Number	15
Learning Ratio	0.8

Also, we compared SRC-TLD with Meanshift, AdaBoost and standard TLD methods under different situations. The results was shown in Fig3. The center location error of our methods are all within 7 pixel accuracy, which is the best compare with the other real-time tracking algorithms.

5.3. Field Experiments

5.3.1. Scenarios

The landing procedure were divided into four sections: (1) The UAV takeoff from the runway. (2) Cruise near the landing area in a large range to test the control system. (c) Cruise near the

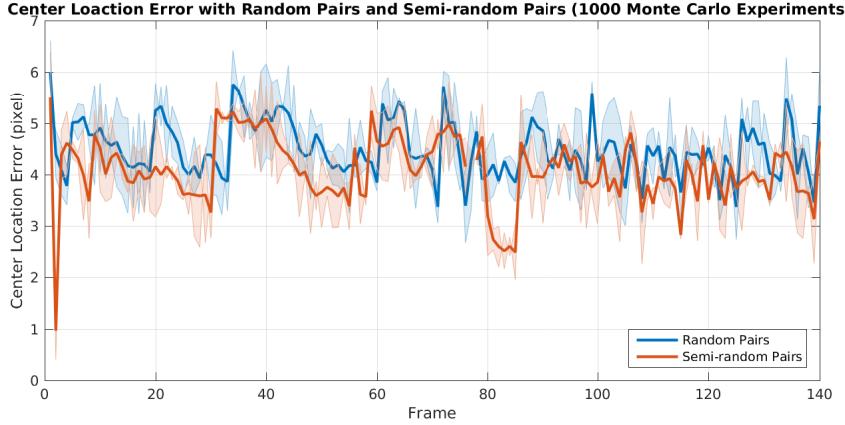


Figure 19: Comparision between random and semi-random coding

landing area in a small range and after the UAV is locked by visual system and received the visual references, the UAV control system was using vision-based localization data and the GPS data was only recorded as the benchmark. (d) Safely landing back to the runway.

5.3.2. Real-flight Experiments

Based on the results of simulation, seven sets of experimental results are conduced to establish the feasibility of the proposed approach. In realistic application, it is very critical requirements that the lateral deviation error from the middle line of the runway and the lateral acceleration of the vehicle should be perfectly eliminated to minimize the damage of the vehicle.

As the theoretical and simulation result discussed, the localization errors in each axis are large when the UAV is far way from the ground visual system. To illustrate the result more clearly, we compared the localization results with DGPS at separated intervals which is shown in Tabel. 5.

Our vision systems uses customized vision algorithms and off-the-shelf hardware to perform in real-time. Actual flight test results on our UAV testbed show our vision-based state estimates are accurate to within 5 cm in each axis of translation and 5 degrees in each axis of rotation.

5.4. Conclusion

This paper presents a complete framework of the ground-based stereo localization problem and shows a real practical use case where the approach presented could be used to guidance the UAV landing autonomously and safely in GNSS-denied scenario. Additional future work will focus on

Table 3: SRC-TLD Method Comparing with Other Tracking Algorithms

Sequence	Frames	AdaBoost	Meanshift	TLD	TLD with Semi-Random Pair
		Center Location Error(pixel) / Speed(fps)			
Small-sized UAV	1	355	6.31 / 9.51	14.39 / 15.35	6.14 / 21.13
	2	384	7.47 / 10.81	14.25 / 15.11	6.37 / 20.91
	3	340	6.61 / 10.14	13.48 / 15.33	5.62 / 21.10
	4	338	7.77 / 10.12	13.85 / 14.71	5.72 / 20.92
Middle-sized UAV	1	137	8.01 / 8.33	11.41 / 13.33	7.85 / 17.00
	2	159	9.65 / 8.59	13.10 / 13.67	6.95 / 16.96
	3	181	8.33 / 8.01	11.13 / 13.03	6.93 / 17.37

Table 4: Eight Experiment Results in Different Weather Condition

No.	Weather Condition	Detection Distance	RMSE i ^{O,c} (m)	RMSE j ^{O,c} (m)	RMSE k ^{O,c} (m)
1	Clear	848.735	<u>0.325</u>	<u>1.451</u>	0.280
2	Clear	892.134	0.239	1.281	0.212
3	Clear	872.311	0.373	1.319	0.282
4	Clear	<u>847.373</u>	0.259	1.401	0.233
5	Clear	857.117	0.242	1.241	<u>0.301</u>
6	Overcast	<u>491.193</u>	<u>0.491</u>	1.676	<u>0.599</u>
7	Overcast	503.175	0.483	1.345	0.576
8	Overcast	534.238	0.470	<u>1.772</u>	0.581

estimate errors over time and investigate methods to improve inevitable error propagation through the inclusion of additional sensors, such as GNSS sensor and on-board IMUs.

References

- [1] V. Kumar and N. Michael, “Opportunities and challenges with autonomous micro aerial vehicles,” *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1279–1291, 2012.

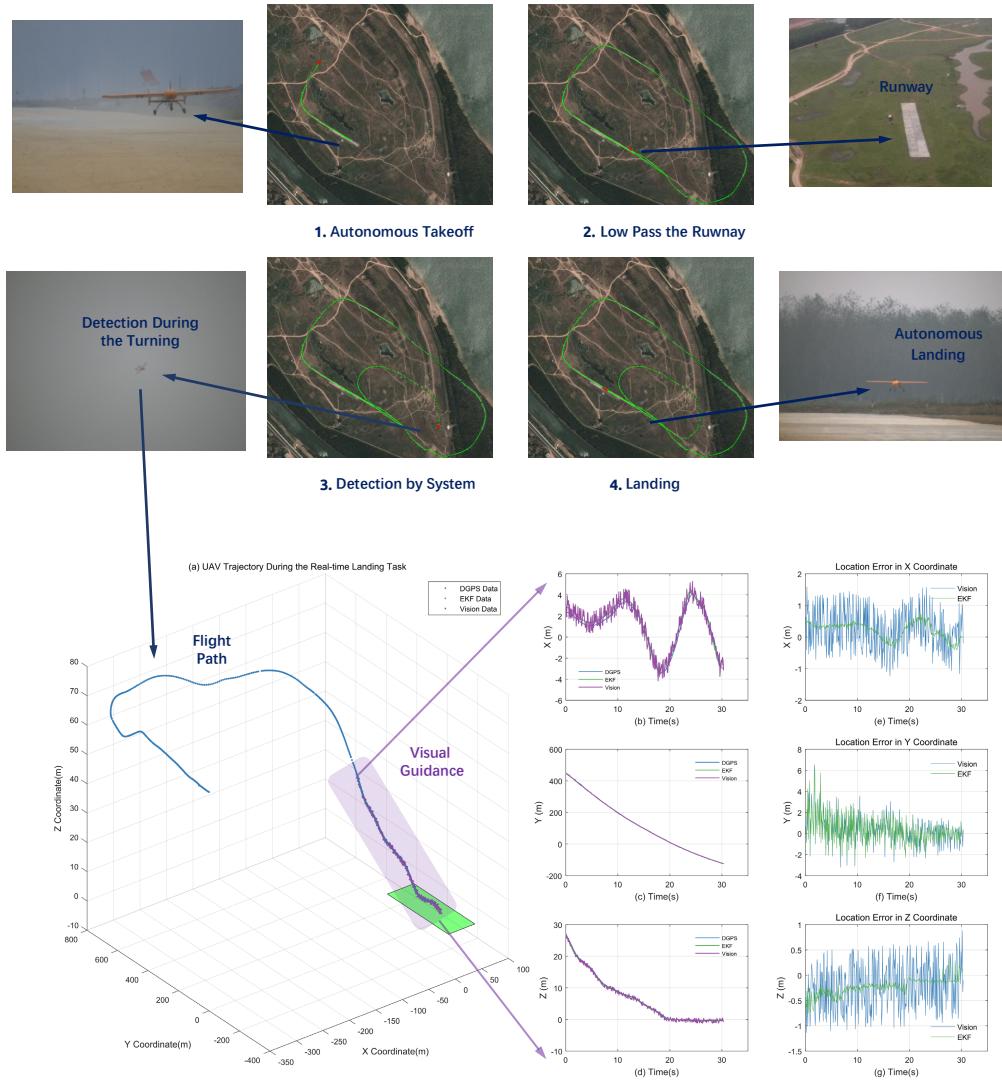


Figure 20: Ground Landing Processure

- [2] K. W. Williams, "A summary of unmanned aircraft accident/incident data: Human factors implications," 2004.
- [3] S. D. Manning, C. E. Rash, P. A. LeDuc, R. K. Noback, and J. McKeon, "The role of human causal factors in us army unmanned aerial vehicle accidents," 2004.
- [4] D. McLean, "Automatic flight control systems(book)," *Englewood Cliffs, NJ, Prentice Hall, 1990, 606*, 1990.
- [5] B. L. Stevens and F. L. Lewis, *Aircraft control and simulation*. John Wiley & Sons, 2003.

Table 5: Guidance Error in Each Axes at Separated Interval

Interval(m)	$i^{O,c}(m)$	$j^{O,c}(m)$	$k^{O,c}(m)$	Interval(m)	$i^{O,c}(m)$	$j^{O,c}(m)$	$k^{O,c}(m)$
600~580	0.332	1.856	0.431	300~280	0.128	1.426	0.171
580~560	0.311	1.434	0.371	280~260	0.123	1.313	0.159
560~540	0.208	1.542	0.300	260~240	0.104	1.104	0.114
540~520	0.182	1.569	0.276	240~220	0.086	0.612	0.129
520~500	0.217	1.863	0.180	220~200	0.099	0.882	0.131
500~480	0.209	1.049	0.216	200~180	0.162	0.933	0.157
480~460	0.185	1.478	0.253	180~160	0.154	0.730	0.136
460~440	0.198	1.245	0.243	160~140	0.134	0.922	0.201
440~420	0.171	1.412	0.151	140~120	0.143	0.696	0.125
420~400	0.186	1.674	0.119	120~100	0.172	0.900	0.118
400~380	0.159	1.679	0.145	100~80	0.139	0.668	0.079
380~360	0.153	1.527	0.197	80~60	0.170	0.696	0.067
360~340	0.171	1.292	0.138	60~40	0.105	0.460	0.055
340~320	0.164	1.360	0.217	40~20	0.126	0.276	0.082
320~300	0.156	1.420	0.204	20~00	0.070	0.261	0.095

Table 6: Three Surface Landing Experiments Results

No.	Weather Condition	Detection Distance	RMSE $i^{O,c}(m)$	RMSE $j^{O,c}(m)$	RMSE $k^{O,c}(m)$
1	Clear	628.735	0.514	1.721	0.371
2	Clear	632.134	0.547	1.363	0.342
3	Clear	<u>622.311</u>	0.583	1.571	0.389

[6] J. Farrell, M. Barth, R. Galijan, and J. Sinko, “Gps/ins based lateral and longitudinal control demonstration: Final report,” *California Partners for Advanced Transit and Highways (PATH)*, 1998.

[7] RUAG, “Opats,” 2015. [Online]. Available: <https://www.ruag.com>

[8] Sierra Nevada, “Dual-thread automatic takeoff and landing system,” 2015. [Online]. Available: <http://www.sncorp.com>

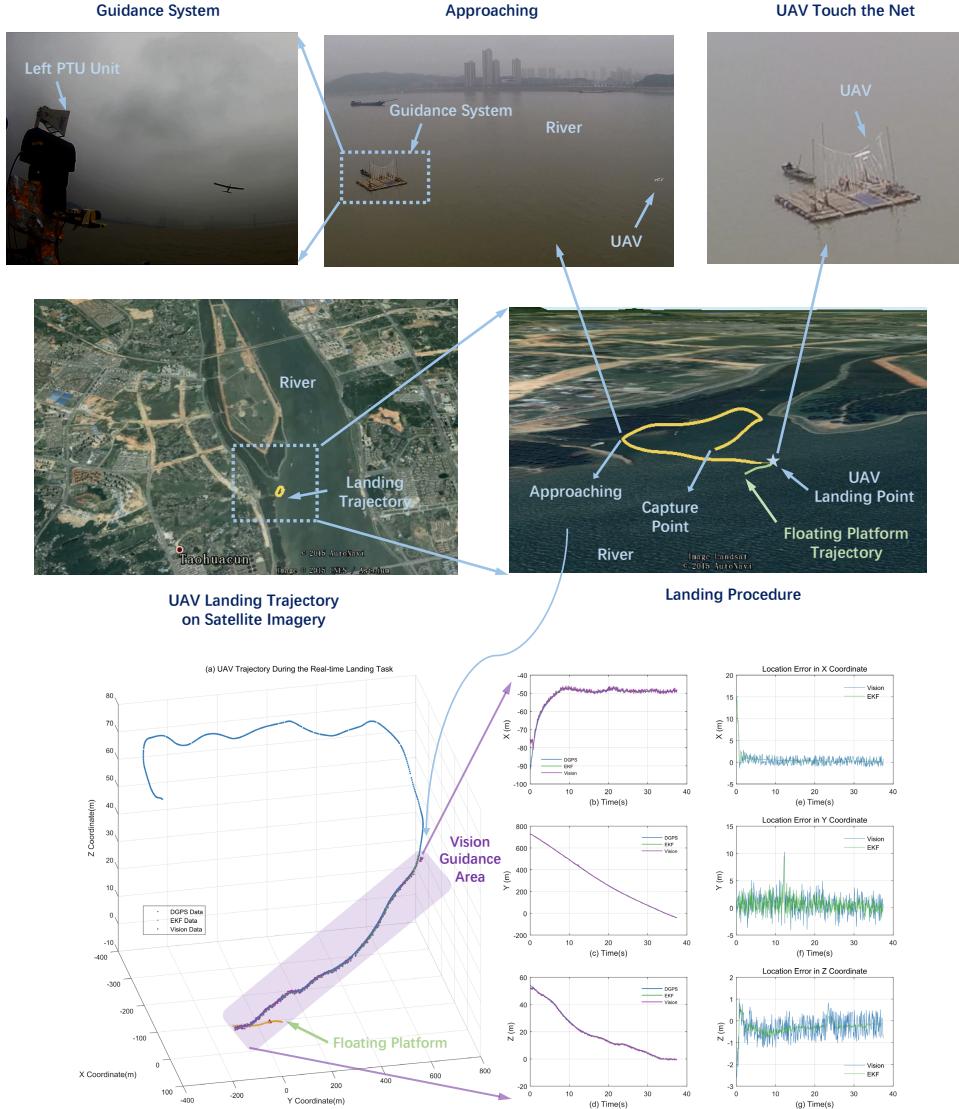


Figure 21: Surface Landing Processure

- [9] ——, “Ucars-v2,” 2015. [Online]. Available: <http://www.sncorp.com>
- [10] M. S. Nolan, “Air Traffic Control,” 2010.
- [11] N. Guenard, T. Hamel, and R. Mahony, “A practical visual servo control for an unmanned aerial vehicle,” *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 331–340, 2008.
- [12] L. Mejías, S. Saripalli, P. Campoy, and G. S. Sukhatme, “Visual servoing of an autonomous helicopter in urban areas using feature tracking,” *Journal of Field Robotics*, vol. 23, no. 3-4,

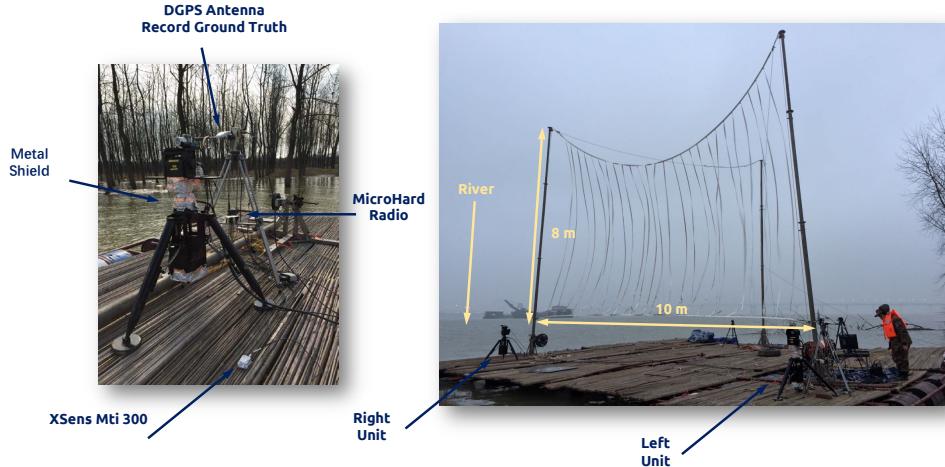


Figure 22: Guidance Configuration on Floating Platform

pp. 185–199, 2006.

- [13] W. Green, P. Oh, and G. Barrows, “Flying insect inspired vision for autonomous aerial robot maneuvers in near-earth environments,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 3. IEEE, 2004, pp. 2347–2352 Vol.3.
- [14] F. Ruffier and N. Franceschini, “Optic Flow Regulation in Unsteady Environments: A Tethered MAV Achieves Terrain Following and Targeted Landing Over a Moving Platform,” pp. 275–293, aug 2014.
- [15] P. Vlantis, P. Marantos, C. P. Bechlioulis, and K. J. Kyriakopoulos, “Quadrotor landing on an inclined platform of a moving ground vehicle,” *Proceedings of the International Conference on Robotics and Automation*, pp. 2202–2207, 2015.
- [16] H. J. Kim, M. Kim, H. Lim, C. Park, S. Yoon, D. Lee, H. Choi, G. Oh, J. Park, and Y. Kim, “Fully autonomous vision-based net-recovery landing system for a fixed-wing UAV,” *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 4, pp. 1320–1333, 2013.
- [17] C. S. Sharp, O. Shakernia, and S. S. Sastry, “A Vision System for Landing an Unmanned Aerial Vehicle (ICRA),” *Proc IEEE Int Conf Robot Autom*, pp. 1720–1727, 2001.

- [18] O. Shakernia, R. Vidal, C. Sharp, Y. Ma, and S. Sastry, “Multiple view motion estimation and control for landing an unmanned aerial vehicle,” in *Proceedings 2002 IEEE International Conference on Robotics and Automation*, vol. 3, no. May. IEEE, 2002, pp. 2793–2798.
- [19] S. Saripalli, J. Montgomery, and G. Sukhatme, “Visually guided landing of an unmanned aerial vehicle,” *IEEE Transactions on Robotics and Automation*, vol. 19, no. 3, pp. 371–380, jun 2003.
- [20] T. S. Richardson, C. G. Jones, A. Likhoded, E. Sparks, A. Jordan, I. Cowling, and S. Willcox, “Automated visionbased recovery of a rotary wing unmanned aerial vehicle onto a moving platform,” *Journal of Field Robotics*, vol. 30, no. 5, pp. 667–684, 2013.
- [21] DLR, “Video autonomous landing at full speed,” 2015. [Online]. Available: http://www.dlr.de/dlr/en/desktopdefault.aspx/tabid-10081/151_read-16413/#/gallery/21679
- [22] M. Garratt, H. POTA, A. LAMBERT, S. ECKERSLEYMASLIN, and C. FARABET, “Visual tracking and LIDAR relative positioning for automated launch and recovery of an unmanned rotorcraft from ships at sea,” *Naval Engineers Journal*, vol. 121, no. 2, pp. 99–9110, 2009.
- [23] W. Wang, G. Song, K. Nonami, M. Hirata, and O. Miyazawa, “Autonomous Control for Micro-Flying Robot and Small Wireless Helicopter X.R.B,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2006, pp. 2906–2911.
- [24] D. Pebrianti, F. Kendoul, S. Azrad, W. Wang, and K. Nonami, “Autonomous hovering and landing of a quad-rotor micro aerial vehicle by means of on ground stereo vision system,” *Journal of System Design and Dynamics*, vol. 4, no. 2, pp. 269–284, 2010.
- [25] C. Martinez, P. Campoy, I. Mondragon, and M. A. Olivares-Mendez, “Trinocular ground system to control UAVs,” in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2009, pp. 3361–3367.
- [26] W. Kong, D. Zhou, D. Zhang, and J. Zhang, “Vision-based autonomous landing system for unmanned aerial vehicle: A survey,” in *Multisensor Fusion and Information Integration for Intelligent Systems (MFI), 2014 International Conference on*. IEEE, 2014, pp. 1–8.

- [27] A. Gautam, P. Sujit, and S. Saripalli, “A survey of autonomous landing techniques for UAVs,” in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, may 2014, pp. 1210–1218.
- [28] A. Betser, P. Vela, and A. Tannenbaum, “Automatic tracking of flying vehicles using geodesic snakes and kalman filtering,” in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 2. IEEE, 2004, pp. 1649–1654.
- [29] R. J. Sattigeri, E. N. Johnson, A. J. Calise, and J.-C. Ha, “Vision-based target tracking with adaptive target state estimator,” 2007.
- [30] A. Blake, M. Isard, *et al.*, *Active contours*. Springer London, 1998, vol. 1.
- [31] R. Malladi, J. Sethian, and B. Vemuri, “Shape modeling with front propagation: a level set approach,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17, no. 2, pp. 158–175, 1995.
- [32] V. Caselles, F. Catte, T. Coll, and F. Dibos, “A geometric model for active contours in image processing,” *Numerische Mathematik*, vol. 66, no. 1, pp. 1–31, Dec 1993.
- [33] Beijing Universal Pioneering Technology Co., Ltd, “iflyuas.” [Online]. Available: <http://www.iflyuas.com/>