JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 39, No. 6, June 2016

Engineering Notes

Monocular Estimation of Ground Orientation for Autonomous Landing of a Quadrotor

John A. Dougherty* and Taeyoung Lee†
George Washington University, Washington, D.C. 20052

DOI: 10.2514/1.G001229

I. Introduction

THE evolution of unmanned aerial vehicle (UAV) technology is progressing at a rapid pace, leading to highly sophisticated systems capable of carrying a wide variety of sensors and payloads. The capability of a vertical-takeoff-and-landing UAV to land autonomously on a surface of arbitrary inclination is an important consideration for a truly autonomous system. To accomplish this, a vehicle must be equipped with sensing to estimate the ground plane orientation and an algorithm for using this information to generate a safe landing trajectory on board in real time. This capability is highly desirable for a UAV operating in unknown terrain to ensure safety in the event of an emergency landing or for a UAV landing on a ship in rough waters experiencing significant deck oscillation. For expensive vehicles and payloads, guaranteeing a safe landing in any reasonable situation is essential.

Landing on an inclined surface is a type of "perching" maneuver in which a vehicle undergoes rapid deceleration, often attaining high angles of attack, to land at a specified point. One proposed application of perching for multirotor vehicles is recharging on power lines [1]. Perching results for fixed-wing vehicles are presented in [2]. The problem of landing a helicopter on an inclined surface is investigated in [3] for the stationary case and in [4] for the dynamic case of landing on a ship in rough sea. Safe landings were achieved in both cases, but no consideration was given to measuring the inclination of the surface during flight. Additionally, autonomous helicopter landing on surfaces inclined up to 60 deg was demonstrated with the help of Velcro® using external navigation by means of a motion capture system.

Recently, there has been research into aggressive maneuvers with quadrotors and landing autonomously on moving and inclined surfaces using onboard and external sensing. Mellinger et al. presented experimental results for quadrotor perching in extreme conditions, such as on vertical and inverted surfaces [6]. The trajectory was treated as a series of segments to reach a goal state with iteratively refined parameters. These perching and aggressive maneuver results are impressive but do not consider the determination of the landing surface during flight. Das et al. proposed a method for measuring surface inclination using inexpensive ultrasonic sensors in real time for landing a quadrotor on an oscillating surface [7]. However, landing results were not presented, and the authors noted that the translational movement in response to changes in quadrotor attitude was not taken into account.

Real-time computer vision has been successfully implemented for the landing of UAVs [8–10]. Simultaneous localization and mapping (SLAM) has been applied to a microhelicopter with a monocular camera [11] and to a quadrotor with front-facing stereo cameras [12] for indoor flight in uncertain and Global-Positioning-System (GPS)-denied environments. Vision-based navigation and control of a quadrotor using onboard sensing has been demonstrated that is capable of performing aggressive maneuvers for landing on inclined surfaces [13,14]. This system requires features to be present at the landing site and on the way to the landing site for localization purposes, and the surface inclination is assumed known. The camera from the Nintendo Wii remote was used in [14] to detect infrared (IR) light-emitting diodes (LEDs) at a landing site, which were placed before flight. As the author suggested, this was a step toward landing without a priori knowledge of the landing site but still required the placement of the IR markers. Fiducial markers were placed on a moving landing platform in [15] to be detected by an onboard camera, allowing a quadrotor to land safely on the moving target, but image processing was done off board. A monocular pose estimation scheme based on IR LEDs mounted on a target object was proposed in [16], which was used to stabilize a quadrotor in various challenging environments. This system was likely the most similar to that proposed in this Note, to the authors' knowledge, with a key difference being that the visual markers were not projected from the quadrotor itself and the system was not used for landing applications but only pose stabilization.

As described previously, many vision-based approaches relying on feature detection impose the unrealistic constraint that a landing site has a known pattern present to be detected by the computer vision system. In most real flight situations, this is not a practical solution. Further, many algorithms suffer from slow computation times, especially on embedded hardware, although this problem is becoming less prevalent with advances in hardware and increasingly efficient computer vision algorithms. One approach free from these limitations was developed in [17], where the slope of the ground surface was extracted directly from the optic flowfield. This purely vision-based approach showed promise in increasing the level of autonomy in landing, although the problem of landing on a sloped surface was not considered.

Previous work presented an onboard measurement system to determine the altitude and surface inclination during flight using a single camera and several low-cost laser pointers [18]. In many cases, a single camera was already on board the quadrotor for another application, such as localization, so there existed the possibility to use the camera for multiple purposes. The geometry model in the previous work was limited to a simplified case where laser beams were assumed to be parallel to the optical axis of the camera. This assumption is eliminated by the three-dimensional model proposed in this Note. Additionally, autonomous landing results are presented that validate the landing trajectory design and simulation.

In this Note, we propose a novel system capable of estimating the distance to and orientation of a static ground plane using inexpensive laser pointers and a single monocular camera mounted on board a quadrotor UAV. The lasers project bright points of light onto the ground surface, which is imaged by the downward-facing camera. An image-processing algorithm detects the positions of the laser dots in the image plane, and a batch-processing estimation procedure provides estimates for the altitude of the vehicle and the normal vector defining the ground plane based on these measurements. The

Received 23 December 2014; revision received 1 September 2015; accepted for publication 8 September 2015; published online 15 April 2016.

*Master's Student, Department of Mechanical and Aerospace Engineering; jadoc@gwu.edu. Student Member AIAA.

†Associate Professor, Department of Mechanical and Aerospace Engineering, 800 22nd St. NW, MAE Suite 3610; tylee@gwu.edu.

image plane coordinates are expressed analytically as functions of known and desired parameters, allowing the Jacobian to be computed explicitly during the estimation procedure. The scheme may be easily generalized to other multirotor vehicles.

The proposed system has several advantages. Since the laser pointers are fixed to the quadrotor, no external features or patterns are required to be present at the landing site; the only requirement is that the projections of the lasers on the ground plane are detectable. By projecting its own "features," a vehicle is able to truly land autonomously in unknown and featureless environments. Additionally, many UAVs are already equipped with a downward-facing monocular camera for SLAM and other applications. Furthermore, the cost of the image processing is minimal, permitting real-time execution, even with modest computational hardware.

The inability to control both the position and orientation of a quadrotor simultaneously complicates the problem of landing trajectory design. A safe trajectory must guide the vehicle toward the landing surface while aligning its attitude to match the inclination of the landing plane. Quadrotor perching has been demonstrated as open-loop maneuvers and online parameter adaptations [6]. In this Note, geometric nonlinear control is used to split the trajectory into two distinct portions: position tracking and attitude tracking [19–21]. In the initial phase, a position-tracking controller guides the quadrotor toward the desired landing position. At a specified switching point, an attitude-tracking controller is engaged to align the quadrotor to the ground surface for a soft touchdown. The entire hybrid trajectory is parameterized by the switching conditions, which are determined to meet boundary conditions for landing. The proposed approach addresses the guidance and control problems simultaneously, and it is developed in a coordinate-free fashion such that it can be uniformly applied to aggressive landing maneuvers.

The major contribution of this Note is a fully autonomous onboard system capable of estimating the altitude of a quadrotor along with the orientation of the static ground plane and using this information to execute a safe landing trajectory. The sensing is a novel combination of a monocular camera and laser pointers, which is a cheap and easily implemented solution. The minimal image processing is executed on board a small computer on module in real time, and the nonlinear least-squares estimation is shown to converge quickly. With knowledge of vehicle altitude and ground plane inclination, a landing trajectory is designed, allowing a quadrotor to land safely on an inclined surface with no prior knowledge of the landing site's inclination.

The quadrotor system model and its notation are described in Sec. II, and the objectives of the landing assistance system are more clearly defined. Section III derives the geometric expressions employed in the estimation procedure and explains both the image-processing routine and the batch-processing scheme. Section IV summarizes the geometric nonlinear control system used and covers the landing trajectory design process, concluding with a numerical example. Experimental results demonstrating the effectiveness of the proposed system are presented in Sec. V, and conclusions follow in Sec. VI.

II. Problem Formulation

A. Quadrotor Dynamic Model

Consider a quadrotor UAV model. This is a system of four identical rotors and propellers located at the vertices of a square, which generate a thrust and torque normal to the plane of this square. We choose an inertial reference frame $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3\}$ and a body-fixed frame $\{\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3\}$. The origin of the body-fixed frame is located at the center of mass of this vehicle. The first and the second axes of the body-fixed frame $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ lie in the plane defined by the centers of the four rotors.

The configuration of this quadrotor UAV is defined by the location of the center of mass and the attitude with respect to the inertial frame. Therefore, the configuration manifold is the special Euclidean group $\mathsf{SE}(3)$, which is the semidirect product of $\mathbb{R}^3$ and the special orthogonal group $\mathsf{SO}(3) = \{R \in \mathbb{R}^{3\times 3} | R^T R = I, \det R = 1\}$.

The mass and the inertial matrix of a quadrotor UAV are denoted by $m \in \mathbb{R}$ and $J \in \mathbb{R}^{3\times 3}$. Its attitude, angular velocity, position, and velocity are defined by $R \in \mathsf{SO}(3), \Omega, x, v \in \mathbb{R}^3$, respectively, where the rotation matrix R represents the linear transformation of a vector from the body-fixed frame to the inertial frame and the angular velocity Ω is represented with respect to the body-fixed frame. The distance between the center of mass to the center of each rotor is $d \in \mathbb{R}$, and the ith rotor generates a thrust f_i and a reaction torque τ_i along $-\boldsymbol{b}_3$ for $1 \le i \le 4$. The magnitude of the total thrust and the total moment in the body-fixed frame are denoted by f, $M \in \mathbb{R}^3$, respectively.

The following conventions are assumed for the rotors and propellers, as well as the thrust and moment that they exert on the quadrotor UAV. We assume that the thrust of each propeller is directly controlled and the direction of the thrust of each propeller is normal to the quadrotor plane. The first and third propellers are assumed to generate a thrust along the direction of $-\boldsymbol{b}_3$ when rotating clockwise; the second and fourth propellers are assumed to generate a thrust along the same direction of $-\boldsymbol{b}_3$ when rotating counterclockwise. Thus, the thrust magnitude is

$$f = \sum_{i=1}^{4} f_i$$

It is positive when the total thrust vector acts along $-\boldsymbol{b}_3$, and it is negative when the total thrust vector acts along $\boldsymbol{b}_3$. By the definition of the rotation matrix $R \in \mathsf{SO}(3)$, the direction of the ith body-fixed axis $\boldsymbol{b}_i$ is given by Re_i in the inertial frame, where $e_1 = [1; 0; 0]$, $e_2 = [0; 1; 0]$, $e_3 = [0; 0; 1] \in \mathbb{R}^3$.

Therefore, the total thrust vector is given by $-fRe_3 \in \mathbb{R}^3$ in the inertial frame. We also assume that the torque generated by each propeller is directly proportional to its thrust. Since it is assumed that the first and the third propellers rotate clockwise and the second and the fourth propellers rotate counterclockwise to generate a positive thrust along the direction of $-\boldsymbol{b}_3$, the torque generated by the ith propeller about $\boldsymbol{b}_3$ can be written as $\tau_i = (-1)^i c_{\tau f} f_i$ for a fixed constant $c_{\tau f}$. All of these assumptions are fairly common in many quadrotor control systems [22,23]. Then, the thrust of each propeller (f_1, f_2, f_3, f_4) is directly converted into f and M, or vice versa. As such, the thrust magnitude $f \in \mathbb{R}$ and the moment vector $M \in \mathbb{R}^3$ are viewed as control inputs throughout this Note.

The equations of motion are given by

$$\dot{x} = v \tag{1}$$

$$m\dot{v} = mge_3 - fRe_3 + \Delta_x \tag{2}$$

$$\dot{R} = R\hat{\Omega} \tag{3}$$

$$J\dot{\Omega} + \Omega \times J\Omega = M + \Delta_R \tag{4}$$

where the hat map $\hat{\cdot}: \mathbb{R}^3 \to \mathfrak{so}(3)$ is defined by the condition that $\hat{x}y = x \times y$ for all $x, y \in \mathbb{R}^3$. This identifies the Lie algebra $\mathfrak{so}(3)$ with $\mathbb{R}^3$ using the vector cross product in $\mathbb{R}^3$. The inverse of the hat map is denoted by the vee map $\vee: \mathfrak{so}(3) \to \mathbb{R}^3$. Unstructured but fixed uncertainties in the translational dynamics and the rotational dynamics of a quadrotor UAV are denoted by Δ_x and $\Delta_R \in \mathbb{R}^3$, respectively.

B. Precise Landing on an Inclined Surface

We wish to land a quadrotor at a point x_L on a flat plane without knowledge of the surface's orientation and altitude. Suppose that the two horizontal components of x_L are known and the normal vector defining the ground plane $\boldsymbol{p}_3$ is unknown but fixed. During flight, $\boldsymbol{p}_3$

and the vertical component of x_L must be measured with onboard sensors. We assume that the position and attitude of the quadrotor can be measured using onboard or external sensors, such as an inertial measurement unit, GPS, or motion-tracking cameras. We propose the following steps to achieve a safe landing:

1) For measurement of landing surface inclination, the quadrotor hovers over the desired landing site and measures both the distance to and orientation of the ground plane, h and $\boldsymbol{p}_3$, respectively, using the proposed onboard laser-vision system. The altitude of the hovering point is chosen sufficiently high to avoid interference or collision with the ground, but it is in the range of sensor capability.

2) For the design of the landing trajectory, a hybrid landing trajectory is designed based on the prior measurements such that the quadrotor lands at x_L with its attitude aligned to the inclined landing surface.

III. Monocular Vision-Based Determination of Landing Surface

In what follows, a framework is developed to employ laser pointers and a single camera to determine the distance to and orientation of a flat landing surface. In Sec. III.A, the geometry is explored to derive an expression relating the positions of the laser dots in the image plane to the distance to and orientation of the ground plane. In Sec. III.B, the image processing required to extract measurements from images is described. Section III.C develops a nonlinear least-squares estimation procedure for batch processing of the measurements, which is applied to a calibration procedure and to the final estimation problem.

A. Geometry of Laser and Camera

Suppose a camera is fixed beneath a quadrotor, pointing toward the ground. Define a camera frame $\{\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3\}$ such that each axis $\boldsymbol{c}_i$ is parallel to the body-fixed axis $\boldsymbol{b}_i$ and its origin Q is located at the aperture of the camera. It is assumed that the optical axis of the camera coincides with $\boldsymbol{c}_3$. Using the pinhole camera model, a (virtual) image plane is parallel to the $\boldsymbol{c}_1 - \boldsymbol{c}_2$ plane and offset from the aperture along $\boldsymbol{c}_3$ by the focal length f_c. In the following discussion, an abstract vector is given in bold (e.g., $\boldsymbol{x}$) and the representation of the vector with respect to a particular reference frame is given in plain font, with a subscript indicating the reference frame (e.g. x_e).

Suppose n laser pointers are installed at points L_j for $j = 1, 2, \ldots, n$ offset from Q along $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ with locations designated by $\boldsymbol{r}_j$, as illustrated in Fig. 1. They point downward but are not necessarily parallel to $\boldsymbol{c}_3$, and the direction of each laser is defined by $\boldsymbol{l}_j$ such that $\|\boldsymbol{l}_j\| = 1$. In this Note, $\boldsymbol{r}_j$ and $\boldsymbol{l}_j$ may be regarded as parameters to be estimated in an offline procedure that is outlined in Sec. III.C. The jth laser beam intersects the ground plane at points P_j, which are photographed by the camera and appear in the image plane. In this subsection, the relation between the location of the laser in the image plane and the distance to the ground plane h, as well as the ground plane's orientation $\boldsymbol{p}_3$, is derived. Recall that $\boldsymbol{p}_3$ is the normal vector defining the ground plane. This derivation is based on the assumption that the quadrotor is hovering directly above x_L; that is, the first two components of x are identical to those of x_L.

For simplicity, we will consider the geometry of a single laser, $n = 1$, but the following development is valid for the jth laser for $j = 1$ to n to n. Let $\gamma_1 \in \mathbb{R}$ be the angle between the vector from L_1 to P_1 (namely, $\boldsymbol{\rho}_{LP,1}$) and the projection of point L_1 to the ground plane along $\boldsymbol{p}_3$, defined as $\boldsymbol{\rho}_{LL',1}$, as shown in Fig. 1b. Notice that

$$\|\boldsymbol{\rho}_{LL',1}\| = \|\boldsymbol{\rho}_{QQ'}\| - r_{1,e}^T p_{3,e} \tag{5}$$

where $\boldsymbol{\rho}_{QQ'}$ is the projection of point Q to the ground plane along $\boldsymbol{p}_3$ such that $\boldsymbol{\rho}_{QQ'} \| \boldsymbol{\rho}_{LL',1}$, and subscript e denotes a vector represented with respect to the inertial frame. Recalling that $\boldsymbol{p}_3$ is orthogonal to the ground plane, we can write the following expression using the triangle created by L_1, L'_1, and P_1:

$$\cos(\gamma_1) = (\|\boldsymbol{\rho}_{QQ'}\| - r_{1,e}^T p_{3,e})(\|\boldsymbol{\rho}_{LP,1}\|)^{-1} \tag{6}$$

Using the dot product, we can also write

$$\cos\gamma_1 = l_{1,e}^T p_{3,e} \tag{7}$$

Define $\zeta \in \mathbb{R}$ as the angle between $\boldsymbol{\rho}_{QQ'}$ and the vector from Q to x_L, given by $h\boldsymbol{e}_3$, where h can be considered the scalar altitude (of the camera frame). From the triangle created by Q, Q', and x_L, we can write

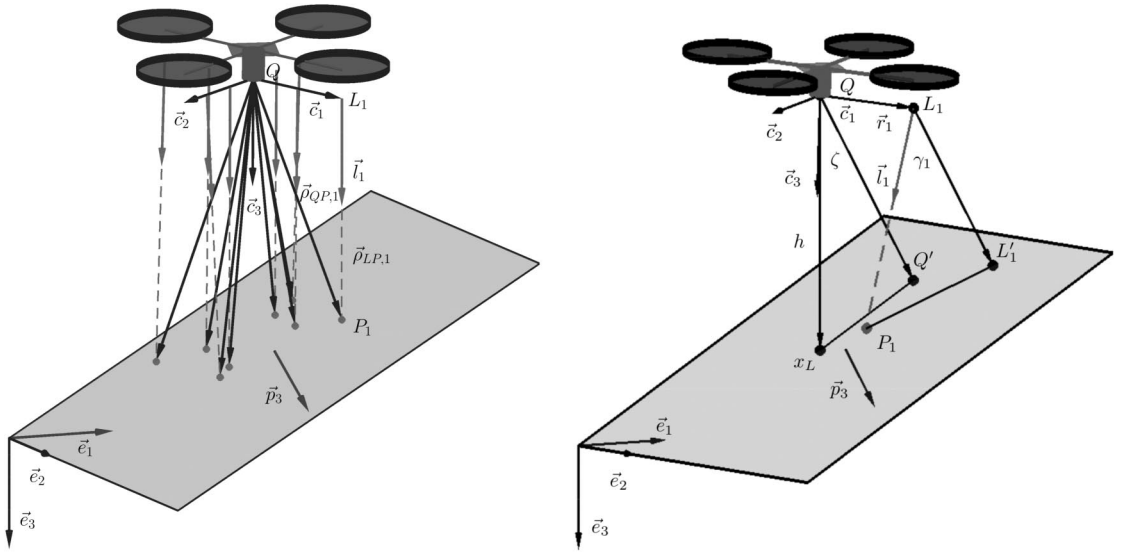
$$\cos(\zeta) = \|\boldsymbol{\rho}_{QQ'}\| h^{-1} \tag{8}$$

Using the dot product, we have

$$\zeta = \cos^{-1}(p_{3,e}^T \boldsymbol{e}_3) \tag{9}$$

Substituting Eq. (9) into Eq. (8) and rearranging, we have

$$\|\boldsymbol{\rho}_{QQ'}\| = h p_{3,e}^T \boldsymbol{e}_3 \tag{10}$$



a) Quadrotor model with eight laser pointers **b) Geometry used in derivation of $\vec{\rho}_{QP,1}$**

Fig. 1 Quadrotor laser geometry.

Substituting Eqs. (7) and (10) into Eq. (6) and rearranging, we have

$$\|\boldsymbol{\rho}_{LP,1}\| = (hp_{3,e}^T \boldsymbol{e}_3 - r_{1,e}^T p_{3,e})(l_{1,e}^T p_{3,e})^{-1} \tag{11}$$

Using the rotation matrix R to rewrite Eq. (11) in terms of known variables and using $\boldsymbol{l}_1$ to get the direction, the representation of the vector from L_1 to P_1 in the camera frame can be expressed as

$$\rho_{LP,1,c} = (hp_{3,e}^T \boldsymbol{e}_3 - r_{1,c}^T R^T p_{3,e})(l_{1,c}^T R^T p_{3,e})^{-1} l_{1,c} \tag{12}$$

where the subscript c denotes the representation of a vector with respect to the camera frame. The vector from Q to P_1 in the camera frame, which is the projection of the first laser dot to the camera aperture, is just the vector sum of $r_{1,c}$ and $\rho_{LP,1,c}$:

$$\rho_{QP,1,c} = r_{1,c} + (hp_{3,e}^T \boldsymbol{e}_3 - r_{1,c}^T R^T p_{3,e})(l_{1,c}^T R^T p_{3,e})^{-1} l_{1,c} \tag{13}$$

Replacing subscript 1 with j, Eq. (13) is valid for the jth laser. This equation describes the location of the laser reflection on the ground in the camera frame in terms of the camera/laser configurations $r_{1,c}$, $l_{1,c}$, the altitude h of the quadrotor, and the ground orientation $p_{3,e}$.

For generality, subscript 1 is now replaced by j. Using the pinhole camera model, the three-dimensional (3-D) coordinates of P_j in the camera frame can be related to the two-dimensional (2-D) image plane coordinates (μ_j, ν_j) as follows:

$$\begin{bmatrix} \mu_j \\ \nu_j \end{bmatrix} = \frac{f_c}{\rho_{QP,j,c}^T \boldsymbol{e}_3} \begin{bmatrix} \rho_{QP,j,c}^T \boldsymbol{e}_1 \\ -\rho_{QP,j,c}^T \boldsymbol{e}_2 \end{bmatrix} \tag{14}$$

where f_c is the camera's focal length, and the negative sign appearing in the ν_j equation corrects for the right-handed coordinate systems employed in this formulation in contrast to the traditional left-handed coordinate system employed in the pinhole camera model. Substituting Eq. (13) and simplifying these equations leads to

$$\begin{bmatrix} \mu_j \\ \nu_j \end{bmatrix} = \frac{f_c}{r_{j,c,3}\alpha_j + l_{j,c,3}\beta_j} \begin{bmatrix} r_{j,c,1}\alpha_j + l_{j,c,1}\beta_j \\ -(r_{j,c,2}\alpha_j + l_{j,c,2}\beta_j) \end{bmatrix} \tag{15}$$

for $j = 1, 2, \ldots, n$, where

$$\alpha_j = l_{j,c}^T R^T p_{3,e}, \qquad \beta_j = hp_{3,e}^T \boldsymbol{e}_3 - r_{j,c}^T R^T p_{3,e} \tag{16}$$

and trailing subscripts 1,2, and 3 indicate first, second, or third vector elements. These equations relate the jth laser dot location (μ_j, ν_j) in the image plane to the unknown altitude h and the ground plane normal vector $\boldsymbol{p}_3$.

B. Image-Processing Algorithm

A simple image-processing routine is required to determine the n pairs of laser dot coordinates from each image. To this end, a simple brightness search is implemented in C to yield a binary image in which only the detected laser dots appear. Each pixel value is tested against some brightness threshold and, since the laser dots appear much brighter than the surroundings (as can be seen in Fig. 2), they are easily detected. Camera parameters such as exposure time and gain can be adjusted to aid in laser detection, and the brightness threshold can be changed to suit different lighting conditions. The contours in this binary image are obtained using OpenCV, and their centroids are calculated and treated as the laser dot coordinates in the 2-D image plane. These coordinates are treated as the measurements in the estimation scheme. See Fig. 2 for an illustration of this complete procedure.

This image-processing scheme is not very computationally expensive and can run on board, even with a small computer on module. The system is tested indoors, but satisfactory results were also obtained outdoors in shaded areas with no modification necessary other than adjusting image-processing parameters; in direct sunlight, however, this approach may need to be modified. The lens used in this work does not result in significant radial or tangential distortion, so the cost of undistorting images is avoided.

C. Nonlinear Least-Squares Estimation

Suppose the aforementioned image-processing algorithm is repeated over a short time period to obtain m image frames. Theoretically, in order to obtain the m distinct frames while holding position, the quadrotor must change its attitude by yawing. In practice, however, small disturbances ensure that the measurements are distinct even with the vehicle simply attempting to hold position. An estimation scheme is necessary to produce an estimate of altitude and ground plane inclination from a set of data consisting of the 2-D location of n laser points in the image plane from each of the m frames. Here, we apply a nonlinear least-squares batch estimator without an a priori state estimate to be implemented at the conclusion of data collection (see [24] Chaps. 1–2 and [25] Chap. 12 for details).

Let $(\tilde{\mu}_j(m), \tilde{\nu}_j(m))$ be the measured laser dot coordinates in the image plane for the jth laser of the mth frame. The measurement vector $\tilde{\boldsymbol{y}} \in \mathbb{R}^{2\,mn}$ becomes

$$\tilde{\boldsymbol{y}} = [\tilde{\mu}_1(1)\ \tilde{\nu}_1(1)\ \cdots\ \tilde{\mu}_n(1)\ \tilde{\nu}_n(1)\ \cdots\ \tilde{\mu}_1(m)\ \tilde{\nu}_1(m)\ \cdots\ \tilde{\mu}_n(m)\ \tilde{\nu}_n(m)]^T \tag{17}$$

This measurement vector contains accumulated measurements that are measured over a certain period of time. Define a state vector $\boldsymbol{x} \in \mathbb{R}^3$ to contain the fixed altitude and two angles representing the orientation of the ground plane, which are the parameters we wish to find for the landing trajectory design. Define $\boldsymbol{\Theta} \in \mathbb{R}^{3n}$ to be a vector containing the laser pointer offset distances and laser directions to be found through a calibration procedure. We can write a measurement model as

$$\tilde{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\Theta}) + \boldsymbol{v} \tag{18}$$

where $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\Theta}) \in \mathbb{R}^{2\,mn}$ is a vector of independent functions relating the state to the measurements populated using Eqs. (15) and (16), and $\boldsymbol{v}$ contains random measurement errors.

Let $\hat{\boldsymbol{y}} = \boldsymbol{f}(\hat{\boldsymbol{x}}, \boldsymbol{\Theta})$ be the estimate of $\boldsymbol{y}$, and $\Delta\boldsymbol{y} = \tilde{\boldsymbol{y}} - \hat{\boldsymbol{y}}$. We wish to minimize

$$J = \frac{1}{2}\Delta\boldsymbol{y}^T W \Delta\boldsymbol{y} \tag{19}$$

where W is a square weighting matrix. Since Eq. (18) is nonlinear, we can start with an initial guess of the state, linearize about that guess, and make successive corrections to the state estimate, which is described in more detail in the following.

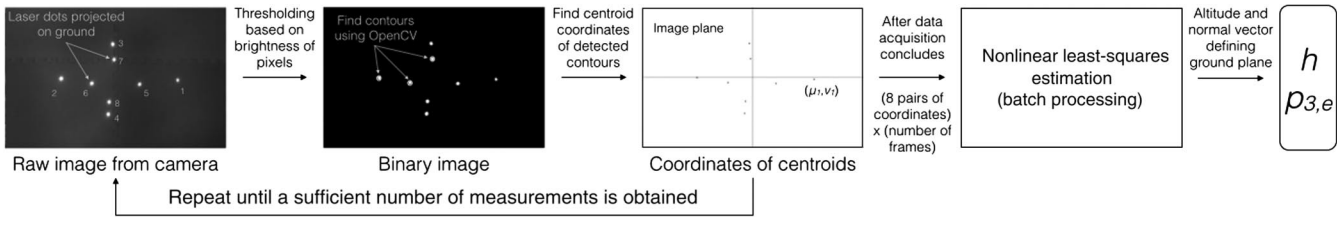


Fig. 2 Outline of scheme to obtain altitude and ground plane orientation from vision measurements.

1. Laser-Vision System Calibration

The accuracy in measuring the distance to and orientation of the ground plane $\boldsymbol{x}$ is dependent on how accurately the laser pointer positions and orientations $\boldsymbol{\Theta}$ are known. Hence, careful consideration must be given to finding $\boldsymbol{l}_j$ and $\boldsymbol{r}_j$ precisely. The objective of the calibration procedure is to find $\boldsymbol{\Theta}$ assuming that $\boldsymbol{x}$ can be obtained by additional sensors available only for calibration. This calibration should be performed offline before flight.

Since $\boldsymbol{l}_j$ is a unit vector, only two constants are needed to represent it. Let $\boldsymbol{l}_j$ be expressed in terms of two angles, δ_j and ϵ_j, which are rotations about the $\boldsymbol{c}_2$ and $\boldsymbol{c}_1$ axes, respectively. Then,

$$l_{j,c} = [\sin(\delta_j) \quad \cos(\delta_j)\sin(\epsilon_j) \quad \cos(\delta_j)\cos(\epsilon_j)]^T \tag{20}$$

Further, since it is assumed that the laser pointers lie in the plane of $\boldsymbol{c}_1$-$\boldsymbol{c}_2$ and are offset along either $\boldsymbol{c}_1$ or $\boldsymbol{c}_2$, only one constant needs to be determined to specify $\boldsymbol{r}_j$. Define $r^*_{j,c}$ to be the nonzero component of $r_{j,c}$ for a particular laser pointer j. Let

$$\boldsymbol{\Theta} = [\delta_1 \quad \epsilon_1 \quad \cdots \quad \delta_n \quad \epsilon_n \quad r^*_{1,c} \quad r^*_{2,c} \quad \cdots \quad r^*_{n,c}]^T \in \mathbb{R}^{3n} \tag{21}$$

Assuming that the attitude and altitude are known at each frame (i.e., $\boldsymbol{x}$ is known), the $2\,mn$ measurements can be processed iteratively to estimate the unknown fixed vectors $\boldsymbol{r}_j$ and $\boldsymbol{l}_j$ for $j = 1, 2, \ldots, n$. An approximation to Eq. (19) is given as

$$J = \frac{1}{2}\Delta\boldsymbol{y}^T W \Delta\boldsymbol{y} \approx J_p \equiv \frac{1}{2}(\Delta\boldsymbol{y}_c - H\Delta\boldsymbol{\Theta})^T W(\Delta\boldsymbol{y}_c - H\Delta\boldsymbol{\Theta}) \tag{22}$$

where $\Delta\boldsymbol{y}_c \equiv \tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\Theta}_c) \in \mathbb{R}^{2\,mn}$, $H \equiv \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\Theta}}\big|_{\Theta_c} \in \mathbb{R}^{2\,mn \times 3n}$ $W \in \mathbb{R}^{2\,mn \times 2\,mn}$ is a weighting matrix, and subscript c denotes the "current" value. Note that the elements of the Jacobian H may be readily obtained by analytically differentiating Eq. (15).

The correction to the guess (see [24] sec. 1.4 for derivation) is given as

$$\Delta\boldsymbol{\Theta} = (H^T W H)^{-1} H^T W \Delta\boldsymbol{y}_c \tag{23}$$

The iterative algorithm requires an initial guess for the unknown fixed $\boldsymbol{\Theta}$ and proceeds using Eqs. (22) and (23), stopping when convergence is detected.

2. Estimating Altitude and Ground Plane Orientation

Once the vectors describing the laser pointers' positions and directions are known from a calibration procedure (i.e., $\boldsymbol{\Theta}$ is known), we can estimate the altitude h and normal vector $\boldsymbol{p}_3$ defining the ground plane. Since $\boldsymbol{p}_3$ is a unit vector, only two constants are needed to specify it. Using spherical coordinates, $p_{3,e}$ may be represented in terms of two angles, θ and ϕ, as

$$p_{3,e} = [\sin(\theta)\cos(\phi) \quad \sin(\theta)\sin(\phi) \quad \cos(\theta)]^T \tag{24}$$

The state vector becomes

$$\boldsymbol{x} = [h \quad \theta \quad \phi]^T \tag{25}$$

and the measurement vector $\tilde{\boldsymbol{y}}$ is Eq. (17). The measurement model is still given as Eq. (18). The approximate cost function is defined as

$$J_p \equiv \frac{1}{2}(\Delta\boldsymbol{y}_c - H\Delta\boldsymbol{x})^T \; W(\Delta\boldsymbol{y}_c - H\Delta\boldsymbol{x}) \tag{26}$$

with

$$\Delta\boldsymbol{x} = (H^T W H)^{-1} H^T W \Delta\boldsymbol{y}_c \tag{27}$$

In this case, $W \in \mathbb{R}^{2\,mn \times 2\,mn}$ and

$$H \equiv \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\bigg|_{x_c} \in \mathbb{R}^{2\,mn \times 3}$$

which is found analytically by differentiating Eq. (15).

The 2 mn measurements are processed iteratively using Eq. (27) to make successive improvements to the current estimate of the fixed, unknown $\boldsymbol{x}$ until convergence is detected or until a stopping criterion is reached. Using Eq. (24), $p_{3,e}$ is then obtained.

IV. Design of Landing Trajectory

A. Quadrotor Tracking Controls

Since it is only possible to achieve asymptotic output tracking for, at most, four quadrotor UAV outputs, we define two flight modes: namely, 1) an attitude-controlled flight mode, and 2) a position-controlled flight mode [19–21] and design control systems for each mode.

Geometric control systems with a generalized integral term to eliminate the effects of disturbances are summarized as follows (see work by Goodarzi et al. [20] for details).

1. Attitude-Tracking Control

Suppose that a smooth attitude command $R_d(t) \in \mathrm{SO(3)}$ satisfying $\dot{R}_d = R_d\hat{\Omega}_d$ is given, where $\Omega_d(t)$ is the desired angular velocity, which is assumed to be uniformly bounded. An attitude error vector $e_R \in \mathbb{R}^3$ and an angular velocity error vector $e_\Omega \in \mathbb{R}^3$ are given by

$$e_R = \frac{1}{2}(R_d^T R - R^T R_d)^\vee, \qquad e_\Omega = \Omega - R^T R_d \Omega_d \tag{28}$$

A nonlinear controller for the attitude controlled flight mode is designed as

$$M = -k_R e_R - k_\Omega e_\Omega - k_I e_I + (R^T R_d \Omega_d)^\wedge J R^T R_d \Omega_d + J R^T R_d \dot{\Omega}_d \tag{29}$$

$$e_I = \int_0^t e_\Omega(\tau) + c_2 e_R(\tau)\,\mathrm{d}\tau \tag{30}$$

where k_R, k_Ω, k_I, and c_2 are positive constants; and the total thrust f can be arbitrarily chosen.

Consider the control moment M defined in Eqs. (29) and (30). If c_2 is sufficiently small, then the equilibrium of the zero attitude-tracking errors $(e_R, e_\Omega, e_I) = (0, 0, \Delta_R/k_I)$ is almost globally asymptotically stable, and the integral term e_I is globally uniformly bounded.

2. Position-Tracking Control

Suppose that an arbitrary smooth position-tracking command $x_d(t) \in \mathbb{R}^3$ is given. The position-tracking errors for the position and the velocity are given by

$$e_x = x - x_d, \qquad e_v = \dot{e}_x = v - \dot{x}_d \tag{31}$$

Similar To Eq. (30), an integral control term for the position-tracking controller is defined as

$$e_i = \int_0^t e_v(\tau) + c_1 e_x(\tau)\,\mathrm{d}\tau \tag{32}$$

for a positive constant c_1. The desired attitude of the position-controlled tracking mode is defined as

$$R_c = [b_{1_c}; b_{3_c} \times b_{1_c}; b_{3_c}], \qquad \hat{\Omega}_c = R_c^T \dot{R}_c \tag{33}$$

where

$$b_{3_c} \in \mathsf{S}^2$$

is given by

$$b_{3_c} = -\frac{-k_x e_x - k_v e_v - k_i \operatorname{sat}_\sigma(e_i) - mge_3 + m\ddot{x}_d}{\| - k_x e_x - k_v e_v - k_i \operatorname{sat}_\sigma(e_i) - mge_3 + m\ddot{x}_d\|} \tag{34}$$

for positive constants k_x, k_v, k_i, and σ. The unit vector $b_{1_c} \in \mathsf{S}^2$ is selected to be orthogonal to b_{3_c}, thereby guaranteeing that $R_c \in \mathsf{SO(3)}$. It can be chosen to specify the desired heading direction. The attitude error and the angular velocity error are defined as Eq. (28) using R_c and Ω_c.

A nonlinear controller for the position-tracking control mode is designed as

$$f = (k_x e_x + k_v e_v + k_i \operatorname{sat}_\sigma(e_i) + mge_3 - m\ddot{x}_d) \cdot Re_3 \tag{35}$$

$$M = -k_R e_R - k_\Omega e_\Omega - k_I e_I + (R^T R_c \Omega_c)^\wedge J R^T R_c \Omega_c + J R^T R_c \dot{\Omega}_c \tag{36}$$

Consider the control system defined by Eqs. (35) and (36). The controller parameters can be chosen such that the zero equilibrium of the tracking errors is almost globally asymptotically stable. The integral terms e_i, and e_I are uniformly bounded.

B. Hybrid Landing Trajectory

The proposed landing trajectory is composed of the position-tracking control mode to translate the quadrotor from the hovering point x_H near the landing point x_L and the attitude-tracking control mode to align the attitude of quadrotor to the landing surface [18]. Due to the almost global stability properties, the control system is robust to switching conditions. The position and the velocity of the quadrotor at the switching instant are defined as $x_S \in \mathbb{R}^3$ and $v_S \in \mathbb{R}^3$, respectively. Let $t = 0$ at the beginning of the landing maneuver, and let t_S and t_L be the times of switching and landing, respectively.

Suppose that t_S and t_L are prescribed and fixed. We define the landing trajectory in terms of x_S and v_S as follows. As the quadrotor is hovering at the beginning of the landing maneuver, the initial conditions are given by

$$x(0) = x_H, \qquad v(0) = 0_{3\times1}, \qquad R(0) = I_{3\times3}, \qquad \Omega(0) = 0_{3\times1} \tag{37}$$

For the given switching condition, we also have $x(t_S) = x_S$ and $v(t_S) = v_S$. The desired landing trajectory satisfying these four boundary conditions and another boundary condition of zero initial acceleration is parameterized by t as follows. For $1 \le i \le 3$, its ith component is given by

$$x_{d_i}(t) = (v_{S_i} t_S + 3(x_{H_i} - x_{S_i}))\frac{t^4}{t_S^4} - (v_{S_i} t_S + 4(x_{H_i} - x_{S_i}))\frac{t^3}{t_S^3} + x_{H_i} \tag{38}$$

When $v_{S_i} = 0$, the preceding trajectory exhibits the additional property that $x_{d_i}(t)$ lies between x_{H_i} and x_{S_i} when $0 \le t \le t_S$. This can be used to guarantee that the quadrotor does not fly below the switching point during its descent for safety.

The desired direction of the first body-fixed axis is chosen such that it is normal to the vertical axis:

$$b_{1_c} = \frac{e_3 \times p_3}{\|e_3 \times p_3\|} \tag{39}$$

Substituting x_d and b_{1_c} into the position-tracking controller [Eqs. (35) and (36)], we numerically simulate the controlled system from the initial condition ([Eq. 37)] until $t = t_S$ to obtain $(x(t), v(t), R(t), \Omega(t))$ for $0 \le t \le t_S$.

Next, when $t > t_S$, the attitude-tracking controller is engaged, where the desired attitude and the desired angular velocity are chosen as

$$R_d(t) = [b_{1_c}, p_{3,e} \times b_{1_c}, p_{3,e}], \qquad \Omega_d(t) = 0_{3\times1} \tag{40}$$

Note that, together with Eq. (39), the preceding definition guarantees that $R_d(t)$ is a rotation matrix in $\mathsf{SO(3)}$. The total thrust f is also selected. Substituting these into Eqs. (28–30), we simulate the controlled system from the initial condition $(x(t_S), v(t_S), R(t_S), \Omega(t_S))$ for $t_S \le t \le t_L$.

In short, the complete landing trajectory is described in terms of the switching position x_S, the switching velocity v_S, and the total thrust f of the attitude-tracking mode. Using these, the problem of designing a landing trajectory is converted into a nonlinear equation to find x_S, v_S, and f such that $x(t_L) = x_L$. The other terminal boundary condition to align the attitude of the quadrotor to the landing surface is not explicitly imposed, as a constraint based on the assumption that $t_L - t_S$ is sufficiently large: the error of the attitude-tracking controller is converged to zero before $t = t_L$. This is not a restrictive assumption, as the attitude dynamics of controlled quadrotors are relatively fast.

As this problem of selecting x_S, v_S, f, t_S, and t_L is underdetermined, one may consider optimizing the parameters with respect to a certain cost function. Instead, these parameters can be chosen according to the following heuristic approach. Note that the landing trajectory can be designed in the two-dimensional plane normal to b_{1_c}. First, the landing time t_L is chosen sufficiently large to avoid a rapid descent, and t_S can be chosen about 0.5 s less than t_L initially. Then, the switching point x_S is selected such that it is on the line joining x_H and x_L, and the distance between x_S and x_L is about one or two times the quadrotor diameter. We can also restrict v_S to be horizontal such that it does not cause any additional downward velocity at t_L. The remaining two parameters (namely, f and the horizontal component of v_S) are iterated numerically such that $x(t_L)$ becomes equal to x_L within the plane normal to b_{1_c}. Finally, the switching time t_S can be further iterated to ensure the attitude error becomes sufficiently small at $t = t_L$. These are illustrated by the following numerical examples.

C. Numerical Example

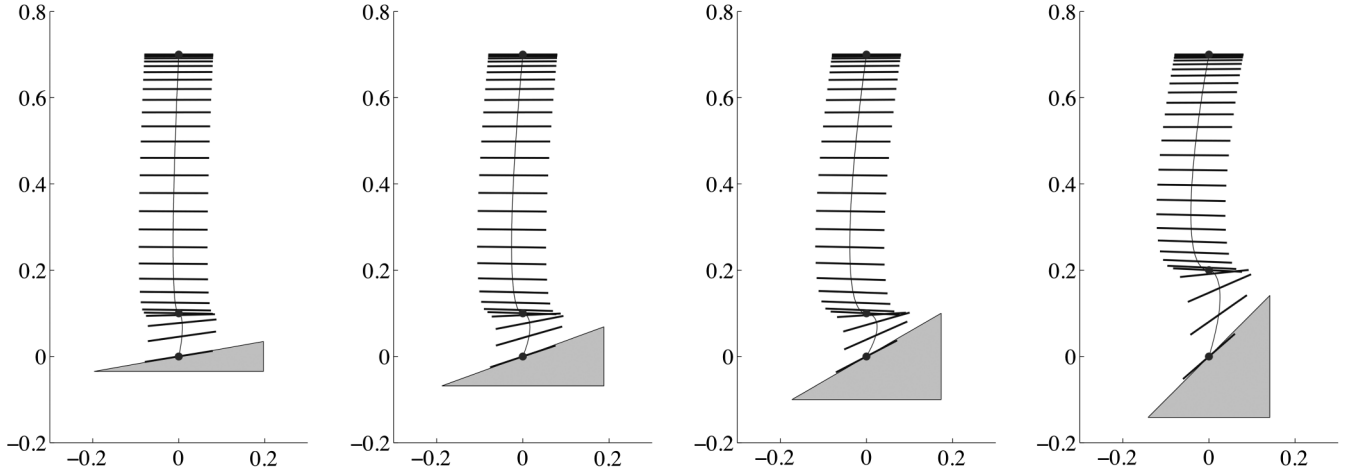
Consider a quadrotor UAV with $m = 1.22$ kg and $J = \operatorname{diag}[0.55, 0.55, 1.05] \times 10^{-2}$ kg · m^2. The hovering point and the landing point are selected as $x_H = [0, 0, 0.7]^T$ m and $x_L = 0_{3\times1}$, respectively. For the given $t_S = 2$ s and $t_L = 2.3$ s, the switching point x_S, the switching velocity v_S, and the total thrust are obtained as discussed previously.

The snapshots that illustrate the resulting position and attitude maneuvers of the quadrotor UAV are shown at Fig. 3 for four landing surface inclinations.

V. Experimental Results

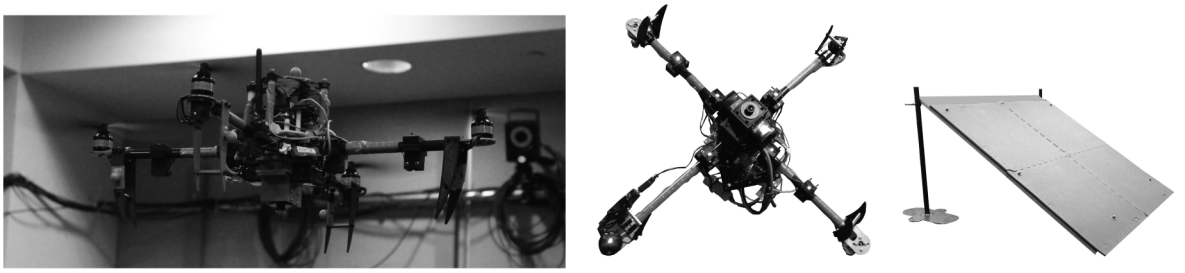
A. Hardware Configuration and Software Structure

We have developed an experimental quadrotor from the ground up using commercially available parts and electronics with some custom additions. The quadrotor is assembled from a Turnigy Talon V2 carbon-fiber frame with a 550 mm span. A Gumstix Overo Air computer on module (COM) runs embedded Linux on board. A multithreaded C code handles flight control, communication, camera control, and image processing. Four Robbe Roxxy brushless dc motors are controlled by BL-Ctrl 2.0 electronic speed controllers, with throttle commands issued by the flight controller via I^{2}C. A custom printed circuit board (PCB) is designed and assembled to handle power distribution and voltage shifting for communication. Position and attitude are obtained using a Vicon motion-capture system and sent to the flight controller wirelessly via XBee; angular velocity from an onboard inertial measurement unit is fused with Vicon data. Keyboard commands can be sent from a host computer to



a) 10° b) 20° c) 30° d) 45°

Fig. 3 Snapshots of simulated landing trajectories for various inclinations.



a) Quadrotor in flight b) Lasers and camera c) Landing platform

Fig. 4 Hardware used in flight experiments.

the quadrotor over WiFi. A 2200 mAh, 11.1 V, lithium-polymer battery powers the quadrotor and its onboard system. The total mass of the vehicle is close to 1.2 kg, fully loaded with the battery.

Eight 650 nm 5 mW laser pointers are mounted to the frame by 3-D-printed plastic mounts in such a way as to maximize the distance between them while having each group of four exit the field of view of the camera at approximately the same altitude. This laser configuration can be seen in Fig. 4b. A Point Grey Research Firefly MV mono camera with a 4 mm lens is rigidly mounted underneath the bottom center plate by a 3-D-printed plastic mount. The focal length of the camera, f_c, is found to be 651.13 pixels by performing an intrinsic camera calibration. The camera streams $752 \times 480 -$ pixel 8-bit monochrome images to the Gumstix via Universal Serial Bus (USB), achieving approximately 16 frames/s with onboard image processing running concurrently with the flight controller. For calibration, when the flight controller is not running, a frame rate of about 32 frames/s is reached. The landing platform that is used in the experiments has an adjustable inclination to approximately 10, 15, 20, 25, and 30 deg, and it is shown in Fig. 4c.

The multithreaded C code reads data from two serial ports that are sent data from the Inertial Measurement Unit (IMU) and XBee, executes flight control, issues throttle commands, accepts wireless keyboard commands from the user, and controls the camera-vision system. The library for controlling IEEE 1394-based cameras, named `libdc1394`, is used to acquire images from the camera. `OpenCV` is used for image processing due to its easy-to-use predefined functions for thresholding and contour detection. The laser-vision system can be run in the background and activated once the quadrotor has reached the appropriate hover position over the landing surface x_H.

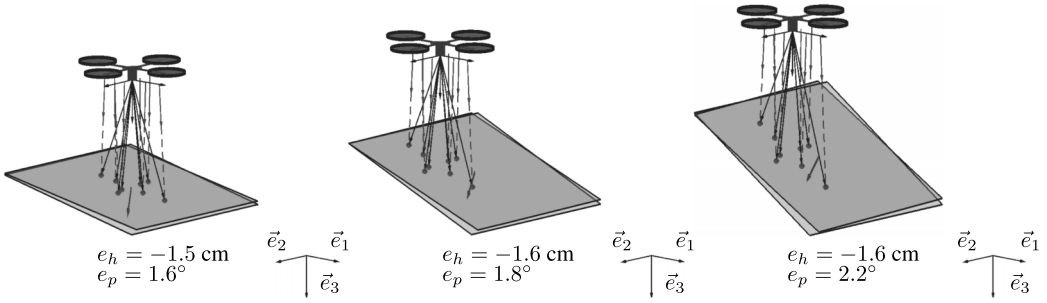
B. Ground Plane Estimation

To collect rich calibration data, the quadrotor was held by hand and moved manually over a flat ground surface. Altitude and attitude data were collected from the Vicon system while vision measurements were being recorded. The calibrated values were then plugged into Eq. (15) along with the Vicon data to calculate the fitted values of μ_j and ν_j to be compared to the experimental measurements and the initial guesses. The calibration procedure results in estimates for $\boldsymbol{r}_j$ and $\boldsymbol{l}_j$ that fit the data quite well. Over 800 frames were analyzed for this calibration procedure to ensure accurate estimates; such a high number of measurements resulted in a long computational time, on the order of 30 min, on a 2.53 GHz MacBook Pro. For this reason, it is unrealistic to perform the estimation for the calibration on board the COM. However, when fewer frames are analyzed, the estimation can easily be performed in a few seconds on board during flight.

To test the accuracy of the laser-vision system in estimating h and $\boldsymbol{p}_3$, several experiments were performed in which the quadrotor acquired measurements while hovering over the inclined landing surface, shown in Fig. 4c. Simultaneously, the quadrotor's position and attitude as determined by the Vicon system were recorded. Subsequently, the position and orientation of the inclined surface were found precisely using the Vicon system. The recorded flight data were processed through the estimation scheme postflight to yield h_{est} and $p_{3,e,\text{est}}$ and compared to the values calculated from the Vicon data: h_v and $p_{3,e,v}$.

For this experiment, three different inclinations were tested: 10, 20, and 30 deg. The quadrotor was piloted to a hovering position over the landing platform such that all the laser dots were projected onto the inclined landing surface at an altitude that was sufficiently high to avoid interference with the ground. Once at the hover position, the vision system was activated and the camera began acquiring images and sending them to the COM for image processing. The image plane coordinates (μ_j, ν_j) extracted from each frame were then written to a file for postprocessing.

The results of the estimation procedure on the experimental flight data are given in Fig. 5. In Figs. 5a, 5c, and 5e, the results are visualized by plotting both the true ground plane and the estimated ground plane. In Figs. 5b, 5d, and 5f, the estimated parameters are



a) 10º inclination **c) 20º inclination** **e) 30º inclination**

	Estimation	Vicon
h, m	0.7353	0.7199
$p_{3,e,1}$	−0.1934	−0.1778
$p_{3,e,2}$	−0.0242	−0.0011
$p_{3,e,3}$	0.9808	0.9841

b) Estimated vs vicon for 10º

	Estimation	Vicon
h, m	0.7382	0.7225
$p_{3,e,1}$	−0.3697	−0.3503
$p_{3,e,2}$	−0.0266	−0.0027
$p_{3,e,3}$	0.9288	0.9366

d) Estimated vs vicon for 20º

	Estimation	Vicon
h, m	0.7930	0.7766
$p_{3,e,1}$	−0.5343	−0.5137
$p_{3,e,2}$	−0.0261	0.0040
$p_{3,e,3}$	0.8449	0.8580

f) Estimated vs vicon for 30º

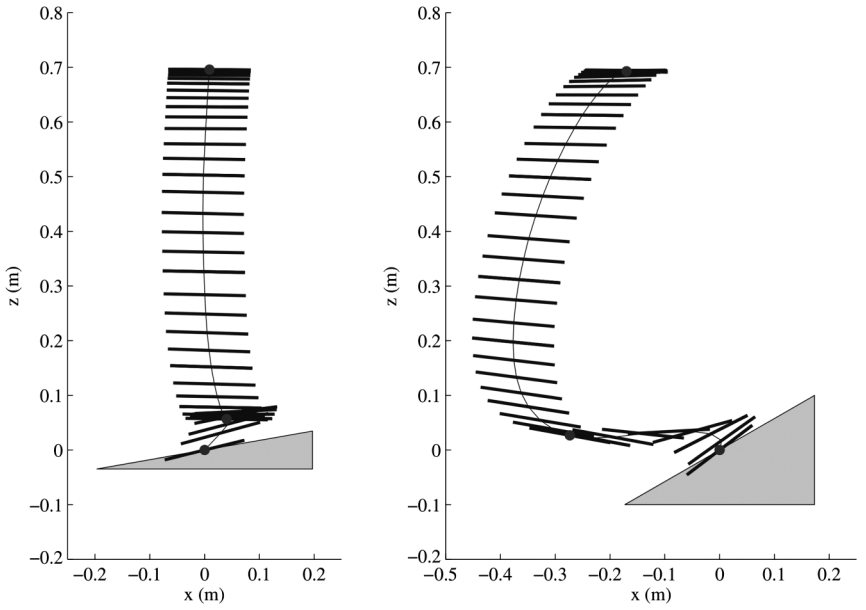
Fig. 5 Estimation results from experimental flight data.

compared to those calculated from the Vicon data. The location and orientation of both the quadrotor and the true ground plane (darker gray) are drawn using the precise Vicon data. The estimated plane (lighter gray) is drawn using the estimated altitude h_{est} and normal vector $p_{3,e,\text{est}}$. The ground plane is considered to be infinite by the estimator, so the limits in the figure are chosen solely for illustration purposes. The position and attitude of the quadrotor in the figures accurately reflect the experimental configuration. Let $e_h = h_v - h_{\text{est}}$ be the altitude error and $e_p = \cos^{-1}(p_{3,e,v} \cdot p_{3,e,\text{est}})$ be the error in calculating the normal vector defining the ground plane. In each case, the system was able to measure h and $\boldsymbol{p}_3$ within an acceptable range of error, with $\|e_h\|_{\max} = 1.6$ cm (or about 2.1%) and $\|e_p\|_{\max} = 2.2$ deg.

C. Autonomous Landing on an Inclined Surface

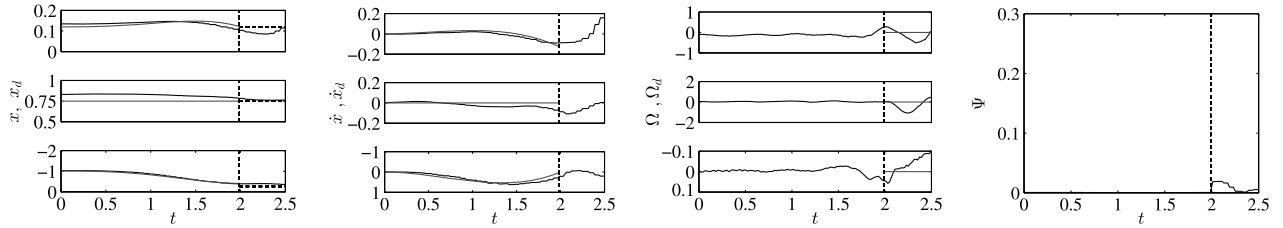
After validating the in-flight measurement and estimation scheme, a number of landing trajectories designed according to the procedure described in Sec. IV were attempted. These trajectories are defined entirely by the switching location x_S, the switching velocity v_S, and the total thrust f of the attitude controlled mode, along with the prescribed switching time and landing time, t_S and t_L, respectively. Experimental results for autonomous landing trajectories are presented for two representative cases, namely, 10 and 30 deg ground plane inclinations. These two cases represent what the authors consider to be a slight and a severe ground plane angle, respectively. Each flight began with a command to fly to the hover position x_H above the desired landing site x_L. This trajectory was hard coded before flight and initiated by a single key press from a pilot at the host computer to start the flight. The quadrotor was then given several seconds to hover for the flight controller to eliminate the steady-state error and move it closer to x_H. At this point, the landing trajectory was initiated by a key press from the pilot. The required parameters for the landing trajectory (x_S, v_S, and f) were chosen before flight based on a numerical simulation and refined through trial and error. For the real flight situation, these parameters could be tabulated and stored in the code to be selected based on the results of the measurement process.

Figure 6 illustrates the experimental landing trajectory results. Thick lines represent the quadrotor's position and attitude during the maneuver as estimated by the Vicon motion capture system. Thin lines represent the trajectory of the center of mass. For the 10 deg case, the switching point x_S was chosen to be on the line connecting x_H to x_L, as was done in the numerical simulation in Sec. IV.C. The

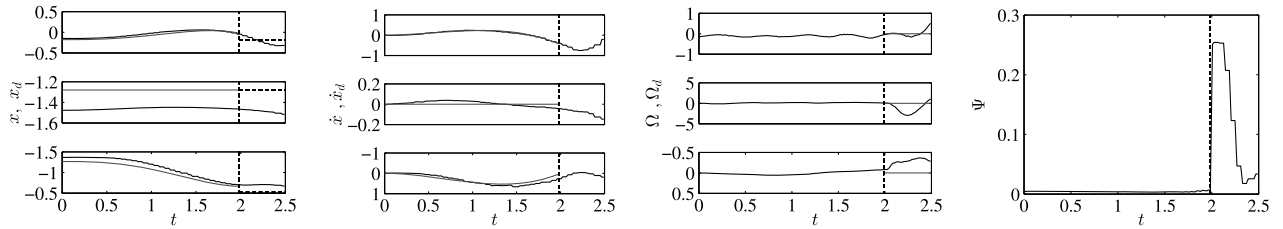


a) 10º **b) 30º**

Fig. 6 Experimental results for two autonomous landing trajectories.



a) x, x_d (m) vs t (s); 10° b) $\dot{x}$, $\dot{x}_d$ (m/s) vs t (s); 10° c) Ω, Ω_d (s^{-1}) vs t (s); 10° d) Ψ vs t (s); 10°



e) x, x_d (m) vs t (s); 30° f) $\dot{x}$, $\dot{x}_d$ (m/s) vs t (s); 30° g) Ω, Ω_d (s^{-1}) vs t (s); 30° h) Ψ vs t (s); 30°

Fig. 7 Data from autonomous landing trajectories with 10 and 30 deg ground plane inclinations.

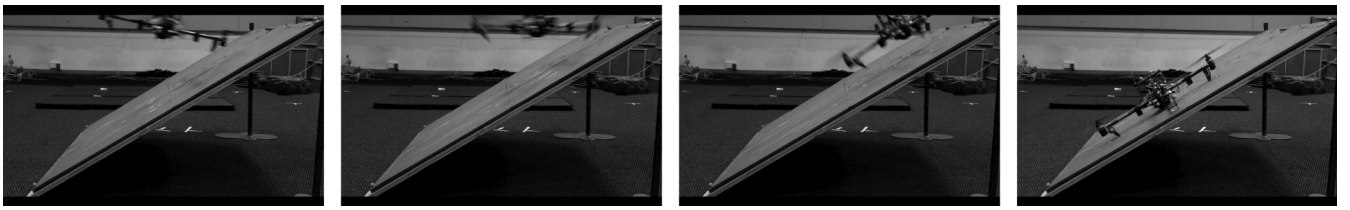


Fig. 8 Landing snapshots for 30 deg inclination.

resulting landing trajectory proved satisfactory, as demonstrated in Fig. 6a; notice the similarity to Fig. 3a.

A similar landing trajectory was attempted for the 30 deg case, where x_S was chosen to be on the line connecting x_H to x_L, shown in Fig. 3c. However, the quadrotor's horizontal velocity near the landing point was too large, causing the vehicle to slide down the surface, since no artificial gripping such as Velcro was used. To reduce the horizontal velocity at touchdown, the desired switching point was moved horizontally by 15 cm, as can be seen in Fig. 6b, resulting in a sufficiently small velocity such that the quadrotor came to rest on the landing surface. Although safe landings were achieved, the nonzero horizontal velocity at touchdown resulted in some error in the quadrotor's final landing position in both the 10 and 30 deg landing attempts.

Flight data from these landing sequences are presented in Fig. 7, plotted as thick black lines. Desired commands are shown as gray solid lines. In Figs. 7a and 7e, x_L is plotted with black horizontal dashed lines. The switching time t_S is indicated in each figure by a black vertical dashed line. Figures 7a and 7e plot the desired position as well as the quadrotor's actual position according to Vicon during the landing sequence. Figures 7b and 7f plot the desired and actual velocities. The tracking performance is satisfactory, even over the short duration of the maneuver, which lasts a total of 2.5 s from the beginning of the descent to touchdown, although a steady-state error exists on the y component in Fig. 7e. Recall that no constraint is imposed on the position and velocity after the attitude-tracking controller is engaged, which is depicted in Fig. 7 with a black vertical dashed line. Conversely, the angular velocity, plotted in Figs. 7c and 7g, is unrestricted during the position-tracking phase but commanded to zero during the attitude-tracking phase, since the desired attitude is constant. The attitude error function Ψ is plotted in Figs. 7d and 7h.

Figure 7 validates the assumption that the controlled attitude dynamics of the quadrotor are relatively fast. At $t = t_S$, the attitude error function Ψ spikes as the desired attitude is set to be aligned to the ground surface. Over the next 0.5 s, the attitude controller reduces Ψ such that it is close to zero at touchdown, when $t = t_L$, as shown in Figs. 7d and 7h. Even for the 30 deg case, Ψ is quickly reduced from about 0.25 to 0.03. Figure 8 shows four snapshots of the latter portion of the landing trajectory for the 30 deg case. In a few 10ths of a second, the quadrotor's attitude becomes aligned to the inclination of the ground surface.

For these landing experiments, a pose estimate is provided by the Vicon system. For a truly autonomous landing, it is unreasonable to use a motion capture system, but onboard pose estimation of the quadrotor is beyond the scope of this Note. Future work should aim to recreate these landing results without any requirement on external sensors.

VI. Conclusions

Inexpensive laser pointers and monocular vision can be employed in a novel sensing system aboard a UAV to estimate the distance to and orientation of a flat, static ground surface during flight. This estimate may be used to design an appropriate landing trajectory based on the switching conditions between the position-tracking and attitude-tracking controller modes. The trajectory may be designed during flight by simply selecting the switching conditions from a table of simulation results for a range of inclinations that are tabulated before flight. Due to the almost global stability properties of each mode of the control system, switching between modes is robust. Contrary to other vision-based approaches, the system does not require features to be present at the landing site. However, the image-processing approach employed here assumes that the intensity of the laser points in the image is much brighter than the rest of the image, which is an assumption that may break down under certain conditions, especially in bright sunlight. The result of the estimation and landing trajectory design is a fully autonomous system capable of operating in unknown environments to aid a UAV in landing safely on an inclined surface.

Acknowledgments

This research has been supported in part by the National Science Foundation under grants CMMI-1243000 (transferred from 1029551), CMMI-1335008, and CNS-1337722.

References

[1] Moore, J., and Tedrake, R., "Powerline Perching with a Fixed-Wing UAV," AIAA Paper 2009-1959, 2009. doi:10.2514/6.2009-1959

[2] Cory, R., and Tedrake, R., "Experiments in Fixed-Wing UAV Perching, " AIAA Paper 2008-7256, 2008. doi:10.2514/6.2008-7256

[3] Moore, J., "A System for Landing an Autonomous Radio Controlled Helicopter on Sloped Terrain," M.S. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, 1994.

[4] Oh, S., Pathak, K., Agrawal, S., Pota, H., and Garratt, M., "Approaches for a Tether-Guided Landing of an Autonomous Helicopter," *IEEE Transactions on Robotics*, Vol. 22, No. 3, 2006, pp. 536–544. doi:10.1109/TRO.2006.870657

[6] Mellinger, D., Michael, N., and Kumar, V., "Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotors," *International Journal of Robotics Research*, Vol. 31, No. 5, 2012, pp. 664–674. doi:10.1177/0278364911434236

[7] Das, P., Swami, S., and Conrad, J., "An Algorithm for Landing a Quadrotor Unmanned Aerial Vehicle on an Oscillating Surface," *Proceedings on the IEEE Southeastcon*, IEEE Publ., Piscataway, NJ, 2012, pp. 1–4.

[8] Sharp, C., Shakernia, O., and Sastry, S., "A Vision System for Landing an Unmanned Aerial Vehicle," *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*, Vol. 2, IEEE Publ., Piscataway, NJ, 2001, pp. 1720–1727. doi:10.1109/ROBOT.2001.932859

[9] Sinopoli, B., Micheli, M., Donato, G., and Koo, T., "Vision Based Navigation for an Unmanned Aerial Vehicle," *Proceedings of the 2001 IEEE International Conference on Robotics and Auto*, Vol. 2, IEEE Publ., Piscataway, NJ, 2001, pp. 1757–1764. doi:10.1109/ROBOT.2001.932864

[10] Saripalli, S., Montgomery, J., and Sukhatme, G., "Visually Guided Landing of an Unmanned Aerial Vehicle," *IEEE Transactions on Robotics and Automation*, Vol. 19, No. 3, 2003, pp. 371–380. doi:10.1109/TRA.2003.810239

[11] Weiss, S., Scaramuzza, D., and Siegwart, R., "Monocular-SLAM–Based Navigation for Autonomous Micro Helicopters in GPS-Denied Environments," *Journal of Field Robotics*, Vol. 28, No. 6, 2011, pp. 854–874. doi:10.1002/rob.20412.

[12] Fraundorfer, F., Heng, L., Honegger, D., Lee, G., Meier, L., Tanskanen, P., and Pollefeys, M., "Vision-Based Autonomous Mapping and Exploration Using a Quadrotor MAV," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE Publ., Piscataway, NJ, 2012, pp. 4557–4564. doi:10.1109/IROS.2012.6385934

[13] Ghadiok, V., Goldin, J., and Ren, W., "On the Design and Development of Attitude Stabilization, Vision-Based Navigation, and Aerial Gripping for a Low-Cost Quadrotor," *Autonomous Robots*, Vol. 33, No. 1, March 2012, pp. 41–68. doi:10.1007/s10514-012-9286-z

[14] Goldin, J. C., "Perching Using a Quadrotor with Onboard Sensing," M.S. Thesis, Utah State Univ., Logan, UT, 2011.

[15] Lee, D., Ryan, T., and Kim, H., "Autonomous Landing of a VTOL UAV on a Moving Platform Using Image-Based Visual Servoing," *Proceedings of the IEEE International Conference on Robotics and Automation*, IEEE Publ., Piscataway, NJ, 2012, pp. 971–976.

[16] Faessler, M., Mueggler, E., Schwabe, K., and Scaramuzza, D., "A Monocular Pose Estimation System based on Infrared LEDs," *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE Publ., Piscataway, NJ, 2014, pp. 907–913.

[17] De Croon, G. C. H. E., Ho, H. W., DeWagter, E., van Kampen, B. R., and Chu, Q. P., "Optic-Flow Based Slope Estimation for Autonomous Landing," *International Journal of Micro Air Vehicles*, Vol. 5, No. 4, 2013, pp. 287–298. doi:10.1260/1756-8293.5.4.287

[18] Dougherty, J., Lee, D., and Lee, T., "Laser-Based Guidance of a Quadrotor UAV for Precise Landing on an Inclined Surface," *Proceedings of the American Control Conference*, IEEE Publ., Piscataway, NJ, 2014, pp. 1210–1215.

[19] Lee, T., Leok, M., and McClamroch, N., "Geometric Tracking Control of a Quadrotor UAV on SE(3)," *Proceedings of the IEEE Conference on Decision and Control*, IEEE Publ., Piscataway, NJ, 2010, pp. 5420–5425.

[20] Goodarzi, F., Lee, D., and Lee, T., "Geometric Nonlinear PID Control of a Quadrotor UAV on SE(3)," *Proceedings of the European Control Conference*, IEEE Publ., Piscataway, NJ, 2013, pp. 3845–3850.

[21] Lee, T., Leok, M., and McClamroch, N., "Nonlinear Robust Tracking Control of a Quadrotor UAV on SE(3)," *Asian Journal of Control*, Vol. 15, No. 2, 2013, pp. 391–408. doi:10.1002/asjc.567

[22] Tayebi, A., and McGilvray, S., "Attitude Stabilization of a VTOL Quadrotor Aircraft," *IEEE Transactions on Control System Technology*, Vol. 14, No. 3, 2006, pp. 562–571. doi:10.1109/TCST.2006.872519

[23] Castillo, P., Lozano, R., and Dzul, A., "Stabilization of a Mini Rotorcraft with Four Rotors," *IEEE Control System Magazine*, Vol. 25, No. 6, 2005, pp. 45–55.

[24] Crassidis, J., and Junkins, J., *Optimal Estimation of Dynamic Systems*, 2nd ed., CRC Press, Boca Raton, FL, 2012, Chaps. 1–2.

[25] Bryson, A., and Ho, Y., *Applied Optimal Control*, Hemisphere, New York, 1975, Chap. 12.