

Vision-Based Local-Level Frame Mapping and Planning in Spherical Coordinates for Miniature Air Vehicles

Huili Yu and Randal W. Beard, *Senior Member, IEEE*

Abstract—This paper presents a vision-based collision avoidance technique for small and miniature air vehicles (MAVs) using local-level frame mapping and planning in spherical coordinates. To explicitly address the obstacle initialization problem, the maps are parameterized using the inverse time-to-collision (TTC). Using bearing-only measurements, an extended Kalman filter is employed to estimate the inverse TTC, azimuth, and elevation to obstacles. Nonlinear observability analysis is used to derive conditions for the observability of the system. Based on these conditions, we design a path planning algorithm that simultaneously minimizes the uncertainties in state estimation while avoiding collisions with obstacles. The behavior of the planning algorithm is analyzed, and the characteristics of the environment in which the planning algorithm is guaranteed to generate collision-free paths for MAVs are described. Numerical results show that the proposed method is successful in solving the path planning problem for MAVs.

Index Terms—Collision avoidance, computer vision, mapping, miniature air vehicles (MAVs), path planning.

I. INTRODUCTION

IN RECENT years, there has been significant research on the development of miniature air vehicles (MAVs) with important military and civilian applications. In situations where a given task might be either too dangerous or difficult, it may be desirable to replace human-operated vehicles with MAVs [1]. For many applications, MAVs are required to navigate in urban environments where obstacles of various types and sizes may hinder the success of the mission. MAVs must have the ability to autonomously avoid collisions with buildings, trees, or other obstacles. Accordingly, path planning is a desirable feature for MAVs, and has received significant attention [1]–[5].

The general framework for the path planning problem can be described as follows: given a description of the environment, find a feasible path between two configurations in the environment that does not result in a collision with any

obstacles. The path planning problem can be grouped into global path planning and local path planning. Global path planning requires complete knowledge of the environment and a static terrain. In that setting, a feasible path from the start to the destination configuration is generated before the vehicle begins its motion [6]. The global path planning problem has been addressed by many researchers, and common solution techniques include potential fields methods, probabilistic roadmap methods, and cell decomposition methods [7].

On the other hand, local path planning is executed in real-time during flight. The basic idea is to first sense the obstacles in the environment and then determine a collision-free path [2]. Local path planning algorithms require sensors to detect obstacles. Among the suite of possible sensors, a video camera is cheap and lightweight and fits the physical requirements of MAVs [2]. Since the camera measurements are obtained in the body frame, it is most natural to create maps and to plan paths directly in the local-level frame of the MAV. References [8]–[11] develop vision-based local-level frame mapping and path planning algorithms. The key idea is to create polar and cylindrical maps and to plan paths using time-to-collision (TTC) and bearing measurements obtained by a camera, without transforming to the inertial frame. However, because of projective geometry, a monocular camera really only measures the bearing to the object. TTC can be estimated by considering the change in the size of the object in the image plane, but this estimate relies on accurately segmenting the image, which can be a noisy process. Therefore, it is a reasonable engineering choice to consider a monocular camera as a bearing-only measurement device and use it to estimate both TTC and bearing. Our previous work in [12] develops an observability-based planning technique for MAVs, where the TTC and bearing are estimated using bearing-only measurements, and a 2-D planning algorithm is developed to minimize the uncertainties in the state estimates while simultaneously avoiding collisions. However, in [12], the obstacle initialization problem was not addressed and the behavior of the planning algorithm was not carefully analyzed.

Feature initialization is a critical issue for a bearing-only camera. Since the camera only provides the bearing to a feature, the TTC estimate for the feature is uncertain when it is initially observed or when the feature exhibits low parallax during motion of the platform. The uncertainties are not well represented by a Gaussian distribution in the context of an extended Kalman filter (EKF) [13]. There have been methods presented for addressing the feature initialization problem in the simultaneous localization and mapping community.

Manuscript received July 12, 2011; revised December 21, 2011; accepted March 1, 2012. Manuscript received in final form March 8, 2012. Date of publication May 1, 2012; date of current version April 17, 2013. This work was supported in part by the Air Force Research Laboratory, Munition Directorate to Scientific Systems Company, Inc., Brigham Young University under SBIR Contract FA 8651-07-C-0094 and Contract FA 8650-08-C-1411, and the OSD and AFRL to SET, Inc. Recommended by Associate Editor T. Parisini.

The authors are with the Department of Electrical and Computer Engineering, Brigham Young University, Provo, UT 84602 USA (e-mail: huiliyu.yhl@gmail.com; beard@byu.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCST.2012.2190604

The methods are classified into delayed and undelayed initialization.

Delayed initialization methods consider the new observed features separately from the map and accumulate depth information over several video frames to reduce the depth uncertainty before adding the new features to the map [14]–[16]. However, the drawback of using these methods for collision avoidance is that the new observed obstacles do not contribute to the path generation until they are added to the map. Reference [13] develops an undelayed feature initialization method that can handle the initialization of features at all depths within the standard EKF framework, using direct parametrization of inverse depth relative to the camera position from which a feature was first observed. The inverse depth parametrization represents a feature by a six-state vector, which is computationally expensive. Once the depth estimate is accurate enough, the inverse depth parametrization is converted to Euclidean XYZ form to speed up the computation.

This paper explores a vision-based local-level frame mapping and planning technique for MAVs. Using bearing-only measurements obtained by a monocular camera, we employ an EKF to estimate the inverse TTC, azimuth, and elevation angles to near-by obstacles, and then construct a map in local-level spherical coordinates. The spherical map is parameterized using the inverse TTC, azimuth, and elevation, which allows the obstacle initialization problem to be addressed explicitly. In addition, compared to using a switching strategy between the inverse depth parametrization and Euclidean XYZ representation, parameterizing the map only using the inverse TTC, azimuth, and elevation enhances the computational efficiency. We perform the observability analysis of the state estimation process from bearing-only measurements and find the conditions for the observability of the system. Based on the derived conditions, we design the planning algorithm that minimizes the uncertainties of the state estimates while simultaneously avoiding collisions with obstacles. We analyze the behavior of the planning algorithm and describe the characteristics of the environment in which the algorithm is guaranteed to generate collision-free paths for the MAV.

This paper is organized as follows. Section II describes our approach to vision-based spherical mapping in the local-level frame using the inverse TTC, azimuth, and elevation. Nonlinear observability analysis of the system is also presented. In Section III, a planning algorithm is presented based on the observability conditions. Section IV analyzes the behavior of the planning algorithm. Section V presents numerical results that demonstrate the effectiveness of the algorithm.

II. VISION-BASED LOCAL-LEVEL FRAME MAPPING IN SPHERICAL COORDINATES

In this section, we will build a map using the inverse TTC to obstacles in the local-level frame of the MAV. The map is constructed in spherical coordinates by estimating the inverse TTC, azimuth, and elevation to obstacles. We then perform a nonlinear observability analysis of the state estimation problem using bearing-only measurements, and find the conditions for complete observability of the system.

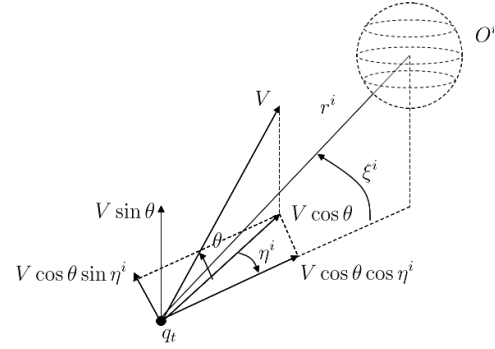


Fig. 1. Motion of the i th obstacle relative to the MAV. The current MAV configuration is q_t . The obstacle is represented by O^i . The pitch angle is represented by θ . The range, azimuth, and elevation to the obstacle are represented by r^i , η^i , and ζ^i . The ground speed is represented by V .

A. Estimates of Inverse TTC, Azimuth, and Elevation

We construct the obstacle map in the local-level frame, where the local-level frame is the body frame with roll and pitch angles removed. The origin of the local-level frame is the MAV's center of mass. The x -axis points out the nose of the airframe when the airframe has a zero pitch angle, the y -axis points out the right wing when the airframe has a zero roll angle, and the z -axis points into the Earth. Throughout this paper, we will assume zero-wind conditions. Let V represent the ground speed of the MAV, and let ψ and θ represent the heading and pitch angles, respectively. With the zero-wind assumption, the ground speed V is equal to the airspeed, which can be measured by a differential pressure sensor. In this paper, we assume that V is available to the estimation and planning algorithms. Fig. 1 shows the motion of the i th obstacle relative to the MAV in the local-level frame, where O^i represents the obstacle, and r^i , η^i , and ζ^i are the range, azimuth, and elevation to the obstacle. The positive directions of azimuth η^i and elevation ζ^i are defined as the right-handed rotation about the z -axis and y -axis of the local-level frame, respectively. Based on Fig. 1, the equations of motion of the obstacle relative to the MAV in terms of the range, azimuth, and elevation are given by

$$\dot{r}^i = -V \cos \theta \cos \eta^i \cos \zeta^i - V \sin \theta \sin \zeta^i \quad (1)$$

$$\dot{\eta}^i = \frac{V \cos \theta \sin \eta^i}{r^i \cos \zeta^i} - \dot{\psi} \quad (2)$$

$$\dot{\zeta}^i = \frac{V \cos \theta \cos \eta^i \sin \zeta^i - V \sin \theta \cos \zeta^i}{r^i} \quad (3)$$

where we assume coordinated turn conditions $\dot{\psi} = g/V \tan \phi$, and where g is the acceleration due to gravity and ϕ is the roll angle of the MAV. Let $\tau^i = V/r^i$ represent the inverse TTC to the obstacle. Substituting τ^i into (1)–(3) and adding process noise gives the equation of motion of the i th obstacle relative to the MAV in the local-level frame in terms of the inverse TTC, azimuth, and elevation as

$$\begin{aligned} \dot{\mathbf{x}}^i &= \mathbf{f}(\mathbf{x}^i, \mathbf{u}) + \mathbf{w}^i \\ &= \begin{bmatrix} (\tau^i)^2 \cos \theta \cos \eta^i \cos \zeta^i + (\tau^i)^2 \sin \theta \sin \zeta^i \\ \tau^i \frac{\cos \theta \sin \eta^i}{\cos \zeta^i} - \dot{\psi} \\ \tau^i \cos \theta \cos \eta^i \sin \zeta^i - \tau^i \sin \theta \cos \zeta^i \end{bmatrix} + \mathbf{w}^i \end{aligned} \quad (4)$$

where $\mathbf{x}^i = [\tau^i, \eta^i, \xi^i]^\top$ is the state, $\mathbf{u} = [\phi, \theta]^\top$ is the controlled input, and the process noise \mathbf{w}^i is a Gaussian random vector with zero mean and covariance matrix \mathbf{Q}^i .

Since the camera directly measures the azimuth and elevation angles, the measurement at time step k is given by

$$\mathbf{z}_k^i = \mathbf{h}(\mathbf{x}_k^i) + \mathbf{v}_k^i = \begin{bmatrix} \eta_k^i \\ \xi_k^i \end{bmatrix} + \mathbf{v}_k^i \quad (5)$$

where the measurement noise \mathbf{v}_k^i is a Gaussian random vector with zero mean and covariance matrix \mathbf{R}_k^i . Based on (4) and (5), the inverse TTC, azimuth, and elevation are estimated using the standard continuous-discrete EKF algorithm [17].

When an obstacle is observed for the first time, we initialize the azimuth and elevation using the measurement data. The uncertainties of the initial inverse TTC to the obstacle can be well approximated by a Gaussian distribution with the mean τ_0 and the standard deviation σ_{τ_0} [13]. The values for τ_0 and σ_{τ_0} are set empirically such that the region within $2\sigma_{\tau_0}$ away from τ_0 spans a range of the TTC from close to the camera up to infinity. Let \mathbf{z}_k^i represent the measurement for the new observed obstacle, and let \mathbf{R}_k^i represent the covariance matrix of measurement noise for that obstacle. The state for the obstacle is initialized as $[\tau_0, \mathbf{z}_k^{i\top}]^\top$, and the error covariance matrix is initialized as

$$\begin{bmatrix} \sigma_{\tau_0}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_k^i \end{bmatrix}.$$

B. Local-Level Frame Mapping in Spherical Coordinates

We build a map directly in the local-level frame instead of the inertial frame. Accordingly, we save the computational expense, and remove the errors associated with transforming the camera data from the local-level frame to the inertial frame, at the expense of updating the map using body motion. Since we do not assume that global positioning system (GPS) is available, we must compute the map in the local-level frame because we do not have knowledge of the inertial location. We construct a map in spherical coordinates, which is more compatible with bearing information obtained by the camera, allowing the data to be processed more efficiently. The maps are constructed using the inverse TTC, azimuth, and elevation, as shown in Fig. 2. The origin of the map is the current location of the MAV. The blue dots are a numerical representation of the 95% uncertainty region in the inverse TTC for each obstacle.

C. Observability Analysis

We use the EKF to estimate the inverse TTC, azimuth, and elevation to obstacles using bearing-only measurements. In order to bound the error covariance computed by the EKF, the system should be observable. We use the nonlinear observability theory developed in [18], which states that the observability is achieved when a system satisfies the nonlinear observability rank condition. Accordingly, we analyze the observability of the system for the i th obstacle given by (4) and (5) by computing the rank of the observability matrix. For an angle α , define $c_\alpha \triangleq \cos \alpha$ and $s_\alpha \triangleq \sin \alpha$. The

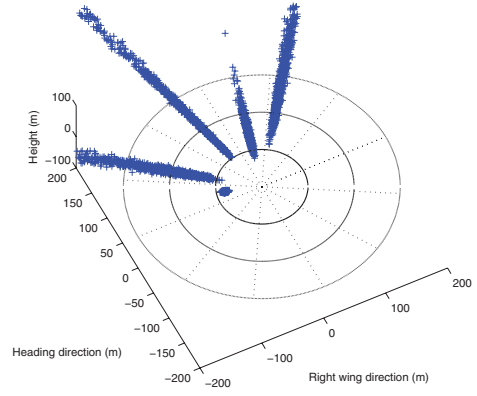


Fig. 2. Local-level frame map in spherical coordinates. The origin of the map is the current location of the MAV. The blue dots are a numerical representation of the 95% uncertainty region in the inverse TTC for each obstacle.

observability matrix is computed using Lie derivatives as described in [18]. The 0th order Lie derivative is

$$L_{\mathbf{f}}^0(\mathbf{h}) = \begin{bmatrix} \eta^i \\ \xi^i \end{bmatrix}. \quad (6)$$

The first-order Lie derivative is given by

$$L_{\mathbf{f}}^1(\mathbf{h}) = \frac{\partial L_{\mathbf{f}}^0(\mathbf{h})}{\partial \mathbf{x}^i} \mathbf{f} = \begin{bmatrix} \tau^i \frac{c_{\theta} s_{\eta^i}}{c_{\xi^i}} - \dot{\psi} \\ \tau^i c_{\theta} c_{\eta^i} s_{\xi^i} - \tau^i s_{\theta} c_{\xi^i} \end{bmatrix}. \quad (7)$$

The second-order Lie derivative is given by

$$L_{\mathbf{f}}^2(\mathbf{h}) = \frac{\partial L_{\mathbf{f}}^1(\mathbf{h})}{\partial \mathbf{x}^i} \mathbf{f} = \begin{bmatrix} \frac{2(\tau^i)^2 c_{\theta}^2 s_{\eta^i} c_{\eta^i}}{c_{\xi^i}^2} - \dot{\psi} \frac{\tau^i c_{\theta} c_{\eta^i}}{c_{\xi^i}} \\ 2(\tau^i)^2 s_{\theta} c_{\theta} c_{\eta^i} s_{\xi^i}^2 - 2(\tau^i)^2 s_{\theta} c_{\theta} c_{\eta^i} c_{\xi^i}^2 \\ + 2(\tau^i)^2 c_{\theta}^2 c_{\eta^i}^2 c_{\xi^i} s_{\xi^i} - 2(\tau^i)^2 s_{\theta}^2 c_{\xi^i} s_{\xi^i} \\ - \frac{(\tau^i)^2 c_{\theta}^2 s_{\eta^i}^2 s_{\xi^i}}{c_{\xi^i}} - \tau^i c_{\theta} s_{\eta^i} s_{\xi^i} \dot{\psi} \end{bmatrix}. \quad (8)$$

The first five rows of the observability matrix are then computed as

$$\mathbf{O}^i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ \frac{c_{\theta} s_{\eta^i}}{c_{\xi^i}} & \frac{\tau^i c_{\theta} c_{\eta^i}}{c_{\xi^i}} & \frac{\tau^i c_{\theta} s_{\eta^i} s_{\xi^i}}{c_{\xi^i}^2} \\ \mathbf{O}_{41}^i & \mathbf{O}_{42}^i & \mathbf{O}_{43}^i \\ \mathbf{O}_{51}^i & \mathbf{O}_{52}^i & \mathbf{O}_{53}^i \end{bmatrix} \quad (9)$$

where

$$\begin{aligned} \mathbf{O}_{41}^i &= c_{\theta} c_{\eta^i} s_{\xi^i} - s_{\theta} c_{\xi^i} \\ \mathbf{O}_{51}^i &= \frac{4\tau^i c_{\theta}^2 s_{\eta^i} c_{\eta^i}}{c_{\xi^i}^2} - \frac{\dot{\psi} c_{\theta} c_{\eta^i}}{c_{\xi^i}} \end{aligned}$$

where it can be shown that subsequent Lie derivatives result in additional rows that do not add to the rank of \mathbf{O}^i , and where the other coefficients \mathbf{O}_{42}^i , etc., do not affect the rank of \mathbf{O}^i . Lemma 1 gives the conditions under which the system for the i th obstacle is observable.

Lemma 1: The i th obstacle, whose motion is given by (4) and (5), is observable at time t if and only if at least one of the following three conditions is satisfied: 1) $\eta^i(t) \neq 0$; 2) $\xi^i(t) \neq \theta(t)$; and 3) $\phi(t) \neq 0$, where $\eta^i(t)$ and $\xi^i(t)$ are the azimuth and elevation angles to the obstacle, and $\phi(t)$ and $\theta(t)$ are the roll and pitch angles of the MAV.

Proof: The observability matrix given by (9) loses rank if and only if all elements in the first column are zero. Accordingly, the i th obstacle, whose motion is given by (4) and (5), is unobservable if and only if all elements in the first column are zero. \mathbf{O}_{31}^i is equal to zero at time t if and only if $\eta^i(t) = 0$. \mathbf{O}_{41}^i is equal to zero at time t if and only if $\xi^i(t) = \theta(t)$. \mathbf{O}_{51}^i is equal to zero at time t if both $\eta^i(t) = 0$ and $\dot{\psi} = g/V \tan \phi = 0$, which implies $\phi(t) = 0$. Therefore, the system is unobservable at time t if and only if all three conditions $\eta^i(t) = 0$, $\xi^i(t) = \theta(t)$, and $\phi(t) = 0$ are simultaneously satisfied. ■

The conditions in Lemma 1 state that the system is observable when the MAV does not directly fly toward the obstacle. When the MAV is not flying directly at the obstacle, parallax can be used to estimate TTC.

III. PATH PLANNING IN THE LOCAL-LEVEL FRAME

The convergence and boundedness of the EKF are achieved when the system is fully observable [19]. Bounds on the EKF error covariance \mathbf{P}^i for the i th obstacle are related to the observability of the system given by Lemma 2 as shown in [19].

Lemma 2 ([19]): Suppose that there exist positive real scalars $\alpha_1, \alpha_2, \beta_1, \beta_2$ such that $\beta_1 \mathbf{I} \leq \mathbf{O}^{i\top} \mathbf{O}^i \leq \beta_2 \mathbf{I}$ and $\alpha_2 \mathbf{I} \geq \mathbf{C}^i \mathbf{C}^{i\top} \geq \alpha_1 \mathbf{I}$ then

$$\left(\frac{1}{\beta_2 + \frac{1}{\alpha_1}} \right) \mathbf{I} \leq \mathbf{P}_k^i \leq \left(\alpha_2 + \frac{1}{\beta_1} \right) \mathbf{I} \quad (10)$$

where \mathbf{C}^i is the controllability matrix and \mathbf{I} is the identity matrix.

The novel result in this paper is the path planning algorithm that minimizes the uncertainties of the inverse TTC, azimuth, and elevation estimates while simultaneously causing the MAV to avoid collisions. Based on Lemma 2, we see that the minimum eigenvalue of the matrix $\mathbf{O}^{i\top} \mathbf{O}^i$ determines the upper bound on the error covariance. To minimize the upper bound on the error covariance, the minimum eigenvalue should be maximized, which is equivalent to minimizing the inverse of the minimum eigenvalue. This implies that minimizing the inverse of the minimum eigenvalue will minimize the uncertainties of the inverse TTC, azimuth, and elevation estimates.

In addition, when the system is unobservable, which corresponds to the case where the MAV directly flies toward the obstacle as shown in Lemma 2, the rank of the observability matrix is two and the inverse of the minimum eigenvalue is infinite. When the system is observable, the rank of the observability matrix is three and the inverse of the minimum eigenvalue is finite. Minimizing the inverse of the minimum eigenvalue ensures that the observability conditions given by Lemma 1 are satisfied, causing the MAV to avoid flying toward the obstacle. Accordingly, the minimization of the inverse of

the minimum eigenvalue will minimize the uncertainties of the state estimation as well as steer the MAV away from the obstacle. Therefore, the minimization of uncertainties and obstacle avoidance are complementary.

Besides collision avoidance, the objective of the planning algorithm is to move the MAV to a goal location, where the MAV requires knowledge of its own inertial position and the inertial position of the goal. Accordingly, for goal seeking, the planning algorithm requires the use of GPS.

Let τ_t^g, η_t^g , and ξ_t^g represent the inverse TTC, azimuth, and elevation to the goal configuration at time t , and let $\mathbf{x}_t^g = [\tau_t^g, \eta_t^g, \xi_t^g]^\top$. Let $\mathbf{x}_t^i = [\tau_t^i, \eta_t^i, \xi_t^i]^\top$ represent the state for the i th obstacle in the MAV's local-level frame map, and let $\mathbf{v}_t = [\mathbf{x}_t^{g\top}, \mathbf{x}_t^{1\top}, \dots, \mathbf{x}_t^{n\top}]^\top$. Let \mathcal{I}_t represent the index set of all n obstacles at time step t , and let τ^l represent the minimum inverse TTC to obstacles that the local planning algorithm reacts to. Let $\mathcal{B}_t = \{i \in \mathcal{I}_t : \tau_t^i \geq \tau^l, \eta_t^i \leq \pi/2, \xi_t^i \leq \pi/2\}$ represent the index set of obstacles with the inverse TTC no less than τ^l and with the azimuth and elevation no greater than $\pi/2$. Define the utility function $S : \mathbb{R}^{3n+3} \rightarrow \mathbb{R}$ as

$$S(\mathbf{v}_t) = \frac{a_1}{(\tau_t^g)^2} + a_2(\eta_t^g)^2 + a_3(\xi_t^g)^2 + \sum_{i=1}^n \frac{b_i I_{\mathcal{B}_t}(i)}{\lambda_{\min}(\mathbf{O}^{i\top} \mathbf{O}^i)} \quad (11)$$

where a_1, a_2, a_3, b_i , and $i = 1, \dots, n$ are positive weights, $\lambda_{\min}(\mathbf{O}^{i\top} \mathbf{O}^i)$ is the minimum eigenvalue of the matrix $\mathbf{O}^{i\top} \mathbf{O}^i$, and $I_{\mathcal{B}_t}(i)$ is the indicator function that is equal to one when $i \in \mathcal{B}_t$ and zero otherwise. Minimizing the first three terms of (11) drives the MAV to the goal. The fourth term penalizes the weighted sum of the inverse of the minimum eigenvalue for all obstacles. By minimizing the fourth term, the algorithm achieves two objectives simultaneously. First, it minimizes the uncertainties in the inverse TTC, azimuth, and elevation estimates. Second, the MAV is steered around the obstacles. We use a look-ahead policy over the horizon T to design the path planner. The cost function to be minimized is given by

$$J = \int_t^{t+T} S(\mathbf{v}_\rho) d\rho \quad (12)$$

subject to the constraints

$$\begin{aligned} \dot{\mathbf{x}}_\rho^g &= \mathbf{f}(\mathbf{x}_\rho^g, \mathbf{u}_\rho) & \dot{\mathbf{x}}_\rho^i &= \mathbf{f}(\mathbf{x}_\rho^i, \mathbf{u}_\rho) \quad i = 1, \dots, n \\ |\phi_\rho| &\leq \phi_{\max}, & |\theta_\rho| &\leq \theta_{\max}. \end{aligned} \quad (13)$$

To solve the constrained optimization problem, we discretize the time horizon T as an m -step look-ahead horizon $\{t, t + \Delta t, \dots, t + m\Delta t\}$, where $\Delta t = T/m$. Equation (12) then becomes

$$J = \sum_{j=1}^m S(\mathbf{v}_{t+j\Delta t}). \quad (14)$$

While a variety of techniques could be used to solve this look-ahead problem, in this paper we solve for the optimal path over the m -step look-ahead horizon by using the nonlinear optimization function `fmincon` in MATLAB [20]. After solving for the optimal path, the first step of the path is followed before resolving the optimization algorithm at the resulting configuration.

IV. ANALYSIS

In this section, we analyze the behavior of the planning algorithm, and describe the characteristics of the environments in which the algorithm is guaranteed to generate collision-free paths for the MAV. We focus our analysis on the environments with spherical obstacles and assume that the locations of the obstacles are perfectly known, i.e., without estimation uncertainties.

To avoid spherical obstacles, we must satisfy

$$\frac{V}{\tau_\rho^i} \geq R_s^i, \quad i = 1, \dots, n \quad \forall \rho \in [t, t+T] \quad (15)$$

where R_s^i is the radius of the i th obstacle. To guarantee collision avoidance, it is necessary to establish a minimum turn away distance d_{\min}^i from each obstacle. The minimum turn away distance is evaluated at the horizontal plane of the MAV's center of mass. Let ϕ_{\max} represent the maximum roll angle of the MAV, and let θ_{\max} represent the maximum pitch angle. The minimum turning radius of the MAV is then given by [21]

$$r_{mt} = \frac{V^2 \cos \theta_{\max}}{g \tan \phi_{\max}}. \quad (16)$$

Lemma 3 establishes the minimum turn away distance for the MAV to avoid a spherical obstacle O^i with radius R_s using the planning algorithm.

Lemma 3: Using the planning algorithm that minimizes the cost function (12) subject to the constraints (13) and (15), avoidance of a collision with a spherical obstacle O^i with the radius R_s is guaranteed if the distance of the MAV to the obstacle satisfies

$$d > \sqrt{(R_s + r_{mt})^2 - r_{mt}^2} - R_s.$$

Proof: Consider the worst case scenario where a MAV is flying perpendicular to a spherical obstacle O^i , and the MAV's center of mass and the center of the obstacle are in the same horizontal plane. Fig. 3 shows the geometry of the worst case scenario projected onto the x - y plane of the inertial frame. The minimum turn away distance from the obstacle d_{\min}^i can be determined when the maximum roll angle and the maximum flight path angle are applied, and the generated circular trajectory with the minimum turning radius r_{mt} is tangent to the boundary of the spherical obstacle. The horizontal plane passing through the tangent point intersects the surface of the obstacle, generating a circle with the radius R_s' . Based on the geometry, the planning algorithm is guaranteed to avoid the obstacle if the distance from the MAV to the obstacle is greater than

$$\sqrt{(R_s' + r_{mt})^2 - r_{mt}^2} - R_s.$$

Since $R_s' < R_s$, we choose

$$d_{\min}^i = \sqrt{(R_s + r_{mt})^2 - r_{mt}^2} - R_s > \sqrt{(R_s' + r_{mt})^2 - r_{mt}^2} - R_s$$

as the minimum turn away distance from the obstacle. Accordingly, if the distance to the obstacle is greater than d_{\min}^i , the optimization problem is feasible and the algorithm will generate a path that avoids the obstacle. ■

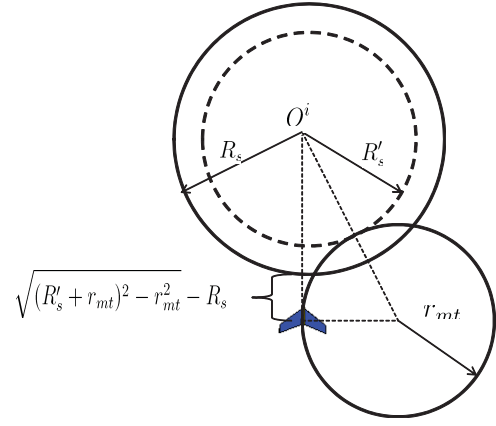


Fig. 3. Worst case scenario where the MAV is flying perpendicular to a spherical obstacle O^i . The MAV's center of mass and the center of the obstacle are in the same horizontal plane.

For the environments with multiple spherical obstacles, we specify the conditions under which the planning algorithm is guaranteed to generate collision-free paths for the MAV. Let \mathcal{C} represent the configuration space. For two configurations $q_1 = [q_{1n}, q_{1e}, q_{1d}, q_{1\psi}, q_{1\theta}]^\top \in \mathcal{C}$ and $q_2 = [q_{2n}, q_{2e}, q_{2d}, q_{2\psi}, q_{2\theta}]^\top \in \mathcal{C}$, where q_{in} , q_{ie} , and q_{id} , $i = 1, 2$, represent North, East, and down (NED) coordinates, and $q_{i\psi}$ and $q_{i\theta}$, $i = 1, 2$, represent the heading and pitch angles, define the 2-D distance between q_1 and q_2 projected on x - y plane of the inertial frame

$$\|q_1 - q_2\|_{2D} \triangleq \sqrt{(q_{1n} - q_{2n})^2 + (q_{1e} - q_{2e})^2}. \quad (17)$$

For a configuration q and the i th obstacle O^i , we define the 2-D distance between q and the boundary of O^i as $d_q^i = \min_{q' \in \partial O^i} \|q' - q\|_{2D}$. Let $d^{ij} = \min_{p_i \in \partial O^i, p_j \in \partial O^j} \|p_i - p_j\|_{2D}$ represent the shortest 2-D distance between the points along the boundaries of the i th obstacle O^i and the j th obstacle O^j . Let q_0 represent the initial MAV configuration and let $q_{O^i} = [q_{O^i n}, q_{O^i e}, q_{O^i d}, q_{O^i \psi}, q_{O^i \theta}]^\top$ represent the configuration of the i th obstacle's center. Let \mathcal{I} represent the index set of obstacles and let $\mathcal{D} \triangleq \{(i, j) \in \mathcal{I} \times \mathcal{I} : |q_{O^i d} - q_{O^j d}| \leq (R_s^i + R_s^j)\}$ represent the set of obstacle pairs in which the altitude difference between each two obstacles is no greater than the sum of their radii. We introduce the notion of local sparseness as Definition 1.

Definition 1: An environment is said to be *locally sparse* if $d^{ij} > \max\{d_{\min}^i, d_{\min}^j\}$, $\forall (i, j) \in \mathcal{D}$.

The local sparseness property of an environment means that each two obstacles in the environment with the altitude difference between them less than the sum of their radii are separated by a distance that is greater than the maximum of their minimum turn away distances. Theorem 1 describes the collision avoidance behavior of the planning algorithm.

Theorem 1: If the environment is locally sparse and the initial MAV configuration satisfies $d_{q_0}^i > d_{\min}^i$, $\forall i \in \mathcal{I}$, then the planning algorithm, which minimizes the cost function (12) subject to the constraints given by (13) and (15), is guaranteed to generate trajectories that will avoid the obstacles for all time t .

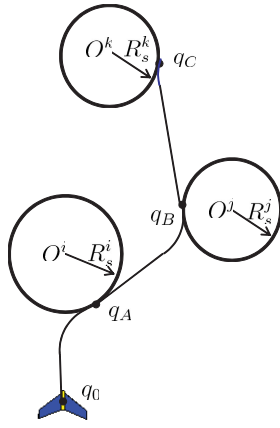


Fig. 4. Planning algorithm maneuvers the MAV to avoid multiple obstacles.

Proof: For the obstacle pairs that are not contained in \mathcal{D} , there exists an altitude gap between each two obstacles. It is obvious that if the planning algorithm can maneuver the MAV to avoid one of the two obstacles then it can maneuver the MAV to avoid the other. We, therefore, focus on collision avoidance for the obstacle pairs that are contained in \mathcal{D} . Consider that the MAV is initially located at q_0 with $d_{q_0}^i > d_{\min}^i, \forall i \in \mathcal{I}$, and that it will collide with an obstacle O^i if it flies along its initial heading, as shown in Fig. 4. Since $d_{q_0}^i > d_{\min}^i$ and the environment is locally sparse, which implies $d^{ij} > \max\{d_{\min}^i, d_{\min}^j\}$, in the worst case scenario the planning algorithm leads to a collision-free path from q_0 to q_A on the boundary of O^i with direction tangent to the boundary, where $d_{q_A}^j > d_{\min}^j$. This means that the MAV certainly has the capability to avoid the obstacle O^j when it reaches q_A . In addition, since $d^{jk} > \max\{d_{\min}^j, d_{\min}^k\}$, in the worst case scenario the planning algorithm leads to a collision-free path from q_A to q_B on the boundary of O^j with direction tangent to the boundary, where $d_{q_B}^k > d_{\min}^k$. This process can be repeated infinitely, which implies that the MAV does not collide with any obstacle for all time t . ■

In Theorem 1, we assumed that the environment is locally sparse in order to provide a theoretical guarantee for collision avoidance behavior of the planning algorithm. However, we note that local sparseness is only a sufficient condition, and that there may be many environments that do not satisfy this condition, but where the MAV may still maneuver without collisions.

V. NUMERICAL RESULTS

The feasibility of the observability-based planning algorithm was tested using a simulation environment developed in MATLAB/SIMULINK, as shown in the subfigures on the right in Fig. 5. The simulator uses a six degree-of-freedom model of the aircraft, where a NED coordinate system is used. The covariance matrix of the process noise for each obstacle was

$$\mathbf{Q}^i = \begin{bmatrix} 0.00001 & 0 & 0 \\ 0 & 0.0001 & 0 \\ 0 & 0 & 0.0001 \end{bmatrix}$$

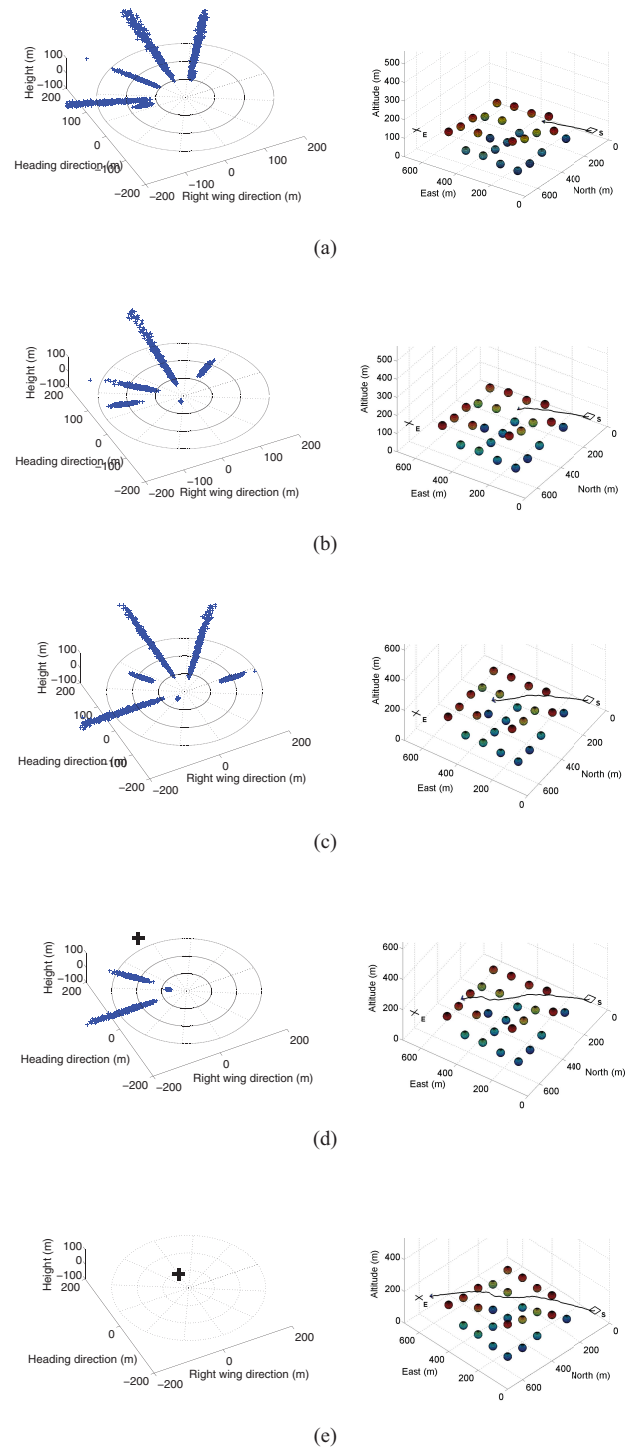


Fig. 5. Evolution of the maps using the inverse TTC parametrization in the local-level frame, and the update of the actual paths followed by the MAV in the inertial frame. (a) $t = 12$ s. (b) $t = 24$ s. (c) $t = 35$ s. (d) $t = 57$ s. (e) $t = 70$ s.

and the covariance matrix of the measurement noise was

$$\mathbf{R}^i = \begin{bmatrix} 0.0012 & 0 \\ 0 & 0.0012 \end{bmatrix}.$$

The values for the initial inverse TTC and its standard deviation were set at $\tau_0 = 0.06$ and $\sigma_{\tau_0} = 0.03$. The groundspeed was $V = 13$ m/s. The maximum roll and pitch angles for

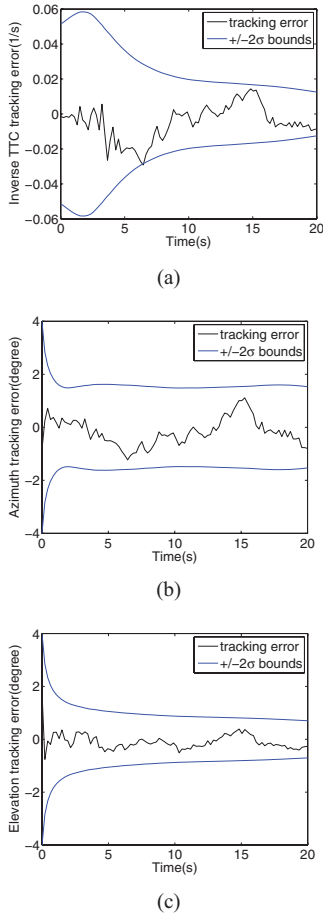


Fig. 6. Tracking error and $\pm 2\sigma$ bounds for the inverse TTC, azimuth, and elevation to the obstacle with x and y coordinates at (150, 250). (a) Inverse TTC tracking error. (b) Azimuth tracking error. (c) Elevation tracking error.

the MAV were 30° and 15° , respectively. The weights were selected as $a_i = 1, \forall i = 1, \dots, 3$ and $b_i = 10, \forall i = 1, \dots, n$. A look-ahead policy over the horizon of 6 seconds was used. We tested the algorithm for multiple obstacle avoidance scenarios. We also conducted Monte Carlo simulations to test the collision avoidance and goal reaching performance of the algorithm with varying measurement uncertainties in the environments with varying minimum 2-D distance between obstacles. The objective of the Monte Carlo simulations is to illustrate that performance degrades gracefully as we relax the zero-noise assumption and the assumption of local sparseness.

A. Multiple Obstacle Avoidance

In the multiple obstacle avoidance scenario, the MAV was commanded to maneuver through twenty-five spherical obstacles between waypoint **S** (0, 100, -20) and waypoint **E** (600, 700, -100), as shown in the subfigures on the right in Fig. 5.

Fig. 5 shows the evolution of the maps using the inverse TTC parametrization in the local-level frame, and the update of the actual paths followed by the MAV in the inertial frame. Subfigures on the left show the local-level frame maps in spherical coordinates. Subfigures on the right show the actual paths. Based on the figures, when the obstacle is first observed, the 95% uncertainty region of the inverse TTC

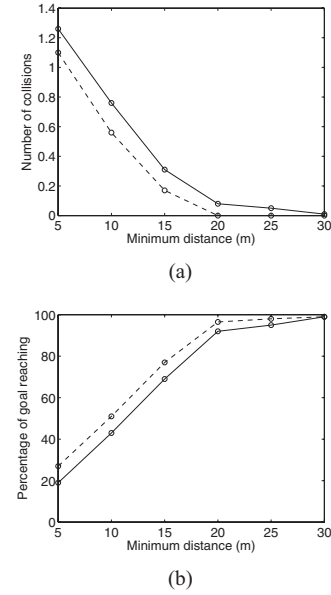


Fig. 7. Statistical performance of the observability-based planning algorithm implemented in environments with varying minimum 2-D distance between obstacles for the cases with and without measurement uncertainties. (a) Average number of collisions over 100 simulation runs versus the minimum 2-D distance between obstacles. (b) Percentage of runs where the MAV reached the goal versus the minimum 2-D distance between obstacles.

includes $\tau = 0$. Accordingly, the uncertainties in the inverse TTC map to the infinity depth. As time progresses, parallax reduces the uncertainties which become progressively smaller, causing the uncertainties in the depth to be reduced. Fig. 6 shows the tracking error and $\pm 2\sigma$ bounds for the inverse TTC, azimuth, and elevation to the obstacle with x and y coordinates at (150, 250).

B. Monte Carlo Simulation

The local sparseness conditions of Theorem 1 are only sufficient for collision avoidance of the local planning algorithm. When the environment is not locally sparse, the algorithm may still generate collision-free paths. Accordingly, we conduct Monte Carlo simulations to demonstrate this is true and to quantify the expected behavior of the algorithm. In addition, in Section IV the analysis of the performance of the planning algorithm does not consider estimation uncertainties in the loop. We, therefore, conduct Monte Carlo simulations to test the performance of the algorithm when the estimation uncertainties are taken into account.

For each environment with a fixed minimum 2-D distance between obstacles, we executed 100 simulation runs. In each simulation run, the MAV was maneuvered from the initial position (120, 120, -60) to the end position (580, 580, -60) through an environment with a fixed minimum 2-D distance. The environment is constructed such that each obstacle is added to the environment based on a uniform distribution over the cubic area with the South-West-down corner (100, 100, -20) and the North-East-up corner (600, 600, -100) until no more obstacle can be added. The radius of all obstacles is 20 m. We evaluate two criteria: the number of collisions and the percentage of runs where the MAV reached the goal.

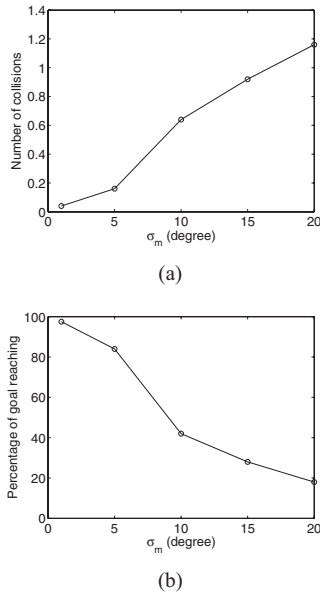


Fig. 8. Statistical performance of the planning algorithm with varying measurement uncertainties. (a) Average number of collisions over 100 simulation runs versus σ_m . (b) Percentage of runs where MAV reached the goal versus σ_m .

We say that the MAV reaches the goal if it is maneuvered to the goal in $t < 100$ seconds without collisions.

Fig. 7 plots the average number of collisions over 100 simulation runs and percentage of runs where the MAV reached the goal versus the minimum 2-D distance between obstacles for the case where the standard deviation for both azimuth and elevation measurement noise is two degrees, and for the case where the obstacle locations are perfectly known. The solid lines show the results for the case with the standard deviation two degrees, and the dashed lines show the results for the case where the obstacle locations are perfectly known. When the standard deviation is two degrees, the number of collisions decreases dramatically as the minimum 2-D distance between obstacles increases from 5 to 20 m. After the minimum distance is greater than 20 m, the number of collisions decreases slowly. Similar phenomenon happens to the percentage of runs where the MAV reached the goal. This is because that more sparse environments cause less collisions. Given $V = 13$ m/s, $\phi_{\max} = 30^\circ$, $\theta_{\max} = 15^\circ$, and $R_s^i = 20$ m, the minimum distance satisfying the local sparseness condition is 19.42 m. Accordingly, when the minimum distance is greater than or equal to 20 m, the environments are locally sparse and the local planning algorithm guarantees collision-free paths if the obstacle locations are perfectly known, which corresponds to the dashed line in Fig. 7(a). Because of the existence of estimation uncertainties, the MAV still encounters a small number of collisions during its flight to the goal, which corresponds to the solid line in Fig. 7(a). In addition, when the minimum distance between obstacles is 10 and 15 m, the average number of collisions is less than one. This shows the local planning algorithm generates collision-free paths for environments that are not locally sparse, and the local sparseness of the environment is only a sufficient condition for collision avoidance behavior of the planning algorithm.

To take into account the effect of estimation uncertainties, we also conduct Monte Carlo simulations to test the performance of the algorithm with varying measurement uncertainties. The minimum 2-D distance between obstacles is fixed at 20 m. Similarly, we evaluate the number of collisions and the percentage of runs where the MAV reached the goal. Let σ_m represent the standard deviation for both azimuth and elevation measurement noise. Fig. 8(a) plots the average number of collisions over 100 simulation runs versus σ_m . Fig. 8(b) plots the percentage of runs where MAV reached the goal versus σ_m . Based on the figure and as expected, as the standard deviation of the measurement noise increases, the number of collisions increases and the percentage of runs where the MAV reached the goal decreases.

VI. CONCLUSION

This paper presents a vision-based local-level frame mapping and planning technique for MAVs operating in unknown environments. To explicitly address the obstacle initialization problem, we construct local-level frame maps in spherical coordinates using the inverse TTC, azimuth, and elevation to obstacles. Using bearing-only measurements, we employ an EKF to estimate the inverse TTC, azimuth, and elevation, and perform an observability analysis of state estimation to find the conditions under which the system was observable. Based on the observability conditions, we design a planning algorithm that minimizes the uncertainties in the state estimation process while simultaneously avoiding collisions with obstacles. We describe the characteristics of the environments in which the planning algorithm was guaranteed to generate collision-free paths for MAVs.

We note that the planning algorithm developed in Section III does not assume that the obstacles were spherical. In order to analyze the algorithm and to make concrete statements about its performance, we made three, admittedly unrealistic, assumptions in Section IV, namely that the obstacles were spherical, that the estimation algorithm returns perfect noise-free estimates of the position of the obstacles, and that the environment was locally sparse. We have shown through Monte Carlo simulations that the performance of the algorithm degrades gracefully when the second two assumptions were relaxed. Since the algorithm itself does not use the spherical assumption, the first assumption is similarly benign.

REFERENCES

- [1] A. Curtis, "Path planning for unmanned air and ground vehicles in urban environments," M.S. thesis, Dept. Elect. Comput. Eng., Brigham Young Univ., Provo, UT, 2008.
- [2] B. Call, "Obstacle avoidance for unmanned air vehicle using computer vision," M.S. thesis, Dept. Comput. Eng., Brigham Young Univ., Provo, UT, Dec. 2006.
- [3] E. Frazzoli, M. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *J. Guid., Control Dyn.*, vol. 25, pp. 116–129, Jan.–Feb. 2002.
- [4] Y. Watanabe, E. Johnson, and A. Calise, "Vision-based approach to obstacle avoidance," in *Proc. AIAA Guid., Nav., Control Conf. Exhibit*, Aug. 2005, no. AIAA-2005-6092.
- [5] A. Pongpunwattana and R. Rysdyk, "Real-time planning for multiple autonomous vehicles in dynamics uncertain environments," *AIAA J. Aerosp. Comput., Inf., Commun.*, vol. 1, pp. 580–604, Dec. 2004.

- [6] K. Sedighi, K. Ashenayi, R. Wainwright, and H. Tai, "Autonomous local path planning for a mobile robot using a genetic algorithm," in *Proc. Congr. Evolut. Comput.*, vol. 2, pp. 1338–1345, Jun. 2004.
- [7] J. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer, 1991.
- [8] H. Yu, R. Beard, and J. Byrne, "Vision-based local multi-resolution mapping and path planning for miniature air vehicles," in *Proc. Amer. Control Conf.*, Jun. 2009, pp. 5247–5252.
- [9] H. Yu, R. Beard, and J. Byrne, "Vision-based navigation frame mapping and path planning for micro air vehicles," in *Proc. AIAA Guid., Nav. Control Conf.*, Aug. 2009, no. AIAA-2009-5679.
- [10] H. Yu, R. Beard, and J. Byrne, "Vision-based navigation frame mapping and planning for collision avoidance for miniature air vehicles," *Special Issue Aerial Robot., Control Eng. Pract.*, vol. 18, no. 7, pp. 824–836, Jul. 2010.
- [11] H. Yu and R. Beard, "Vision-based 3-D navigation frame mapping and planning for collision avoidance for micro air vehicles," in *Proc. AIAA Guid., Nav., Control Conf.*, Aug. 2010, no. AIAA-2010-8208.
- [12] H. Yu, R. Sharma, R. Beard, and C. Taylor, "Observability-based local path planning and collision avoidance for micro air vehicles using bearing-only measurements," in *Proc. IEEE Amer. Control Conf.*, Jul. 2011, pp. 4649–4654.
- [13] J. Civera, A. Davision, and J. Montiel, "Inverse depth parametrization for monocular SLAM," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 932–945, Oct. 2008.
- [14] M. Bryson and S. Sukkarieh, "Bearing-only SLAM for an airborne vehicle," in *Proc. Austral. Conf. Robot. Autom.*, Sydney, Australia, 2005, pp. 1–9.
- [15] A. Davision, "Real-time simultaneous localisation and mapping with a single camera," in *Proc. Int. Conf. Comput. Vis.*, vol. 2, Nice, France, Oct. 2003, pp. 1403–1410.
- [16] J. H. Kim and S. Sukkarieh, "Airborne simultaneous localisation and map building," in *Proc. IEEE Int. Conf. Robot. Autom.*, Sep. 2003, pp. 406–411.
- [17] F. Lewis, *Optimal Estimation: With an Introduction to Stochastic Control Theory*. New York: Wiley, 1986.
- [18] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Trans. Autom. Control*, vol. 22, no. 5, pp. 728–740, Oct. 1977.
- [19] Y. Song and J. W. Grizzle, "The extended Kalman filter as a local asymptotic observer for discrete-time nonlinear system," *J. Math. Syst., Estim., Control*, vol. 5, no. 1, pp. 59–78, 1995.
- [20] *MathWorks, Optimization Toolbox* [Online]. Available: <http://www.mathworks.com/>
- [21] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and Practice*. Princeton, NJ: Princeton Univ. Press, 2011.



Huili Yu received the Ph.D. degree in electrical and computer engineering from Brigham Young University, Provo, UT, in 2011.

His current research interests include vision-based localization and mapping, path planning, collision avoidance, cooperative controls, estimation, robotics, and miniature air vehicles.



Randal W. Beard (SM'03) received the B.S. degree in electrical engineering from the University of Utah, Salt Lake City, in 1991, the M.S. degree in electrical engineering, the M.S. degree in mathematics, and the Ph.D. degree in electrical engineering, from Rensselaer Polytechnic Institute, Troy, NY, in 1993, 1994, and 1995, respectively.

He has been with the Electrical and Computer Engineering Department, Brigham Young University, Provo, UT, since 1996, where he is currently a Professor. From 1997 to 1998, he was a Summer

Faculty Fellow with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena. From 2006 to 2007, he was a National Research Council Fellow with the Air Force Research Laboratories, Eglin Air Force Base, Fort Walton Beach, FL. His current research interests include autonomous systems, unmanned air vehicles, multiple vehicle coordination and controls, and vision-based guidance and control algorithms for micro-air vehicles.

Dr. Beard is currently an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL. His students have won numerous competitions and awards for their work on micro-air vehicles.