

Vision-Based State Estimation for Autonomous Micro Air Vehicles

Thomas P. Webb* and Richard J. Prazenica[†]
University of Florida, Shalimar, Florida 32579

Andrew J. Kurdila[‡]

Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061

and

Rick Lind[§]

University of Florida, Gainesville, Florida 32611

DOI: 10.2514/1.22398

The realization of vision-based, autonomous flight for micro air vehicles presents numerous technical challenges in diverse fields such as image processing, trajectory planning, control theory, and microhardware design. This paper explores one aspect of the overall problem: robust, real-time estimation of aircraft states from a set of tracked feature points. This vision-based approach is realized via an implementation of the implicit extended Kalman filter, a variation of the classical Kalman filter that allows for nonlinear dynamic models and updates from measurements that are implicit functions of the state variables. The epipolar constraint, which is a geometric relationship between the feature point position vectors and the camera translation vector, is employed in this paper as the implicit measurement in the Kalman filter. In order for the vision-based Kalman filter to provide reliable state estimation, it must be robust with respect to modeling errors in the dynamic propagator. This is especially true because the aerodynamic forces and moments acting on a micro air vehicle are frequently not known to a high degree of accuracy. Furthermore, the algorithm must be robust with respect to noise in the measured feature point positions in the image plane. This paper explores both of these robustness issues using results from a micro air vehicle simulation model developed at the NASA Langley Research Center. In particular, a hierarchy of dynamic models, ranging from a random walk model to a high-fidelity nonlinear micro air vehicle model, is employed in the Kalman filter for a simulated micro air vehicle trajectory with varying levels of measurement noise. It is demonstrated that the vision-based measurement updates in the filter are capable of compensating for significant modeling errors and filter initialization errors. As would be expected, superior overall results are achieved using higher-fidelity dynamic models in the Kalman filter. The work presented in this paper represents the first step toward the ultimate objective of incorporating vision-based state estimation into the design of autonomous flight control systems for micro air vehicles operating in urban environments.

Introduction

RESEARCHERS have studied several important tasks related to the development of agile and autonomous micro air vehicles (MAVs) in the past few years. Some of these vehicles have a wing span of less than 6 in. and can carry payloads that are measured in grams. Substantial progress has been made in the fabrication, structural design, and development of inner-loop controllers that use vision-based, horizon-tracking algorithms [1–3]. Still, there remain substantial technical barriers that must be addressed before this class of vehicles is capable of vision-based, autonomous flight. The task of flying autonomous MAVs in environments confined by buildings and trees with additional features from ground vehicles and civilians,

as well as poor weather, requires a host of innovations in vision-based flight control. Active vision-based control of agile autonomous vehicles in complicated three-dimensional urban environments requires fundamental and ground-breaking innovations in multiple, related disciplines such as control theory, vision processing, trajectory planning, microhardware for computation, control and sensing, and MAV design and fabrication. In this paper, the problem of autonomous, vision-based flight for MAVs is not addressed in all its generality. Rather, this paper focuses on a specific enabling technology: vision-based state estimation. Specifically, this paper explores state estimation algorithms that synthesize vision-based measurements, in the form of tracked feature points provided by image processing, with a vehicle dynamic model in a Kalman filtering framework.

The use of vision-based observations for state estimation is attractive for several reasons, especially because the objective is autonomous flight in urban environments. These environments are characterized by many diverse features which can provide useful data for vision-based flight. In particular, these visual data are essential for obstacle avoidance and the more general problem of path planning. It is convenient to use the same information to estimate the state, or motion, of the aircraft. MAVs, such as those described in [1,4], are commonly instrumented with a single camera and a global positioning system (GPS) receiver. Therefore, vision-based state estimation, augmented with GPS, places no additional hardware requirements on typical MAVs. Although GPS is a useful source of navigation data, it can be subject to dropout in dense urban environments. Therefore, it is desirable to employ a vision-based state estimator that can incorporate GPS data when available, but is

Presented as Paper 5349 at the AIAA Guidance, Navigation, and Control Conference and Exhibit, Providence, RI, 16–19 August 2004; received 11 January 2006; revision received 25 May 2006; accepted for publication 27 September 2006. Copyright © 2006 by the authors. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/07 \$10.00 in correspondence with the CCC.

*Ph.D. Candidate, Research in Engineering Education Facility; thomas.webb2@eglin.af.mil. Member AIAA.

[†]Visiting Assistant Professor, Research in Engineering Education Facility; prazenic@ufl.edu. Member AIAA.

[‡]W. Martin Johnson Professor, Department of Mechanical Engineering; kurdila@vt.edu. Associate Fellow AIAA.

[§]Assistant Professor, Department of Mechanical and Aerospace Engineering; ricklind@ufl.edu. Senior Member AIAA.

also capable of operating without the benefit of GPS updates for extended periods of time. The fusion of GPS and visual data for optimal state estimation is a topic for future research and is not treated in this paper. Here it is simply noted that, at least in principle, GPS data can be incorporated into the framework of the vision-based Kalman filter in a straightforward manner.

The classical Kalman filter has been widely applied for robust, real-time state estimation for aerospace systems. In principle, the filter integrates information from sensor measurements with predictions from a dynamic state-space model in order to compute optimal state estimates [5,6]. Although the Kalman filter was initially derived for linear systems, a variation known as the extended Kalman filter was later developed to consider nonlinear dynamic models. The standard implementation of the Kalman filter includes a measurement update in which the measurement is an *explicit* function of the states. The objective of this paper is to investigate the use of tracked feature points for state estimation, which gives rise to measurements that are *implicit* functions of the states. In a series of papers, Soatto et al. [7,8] derived rigorous forms of the implicit extended Kalman filter (IEKF) that incorporate implicit, vision-based measurements into the framework of the extended Kalman filter. Soatto et al. [7] first applied an IEKF implementation that makes use of the epipolar constraint, also termed the essential or coplanarity constraint. This constraint is a geometric expression of the fact that the position vectors of a static feature point relative to the camera at two instants in time are coplanar with the camera translation vector. Soatto et al. [7] used this approach to characterize the motion of a cloud of feature points about a fixed camera. A first-order random walk was employed as the dynamic model in the filter. Later, Soatto and Perona [8] employed an IEKF algorithm based on the subspace constraint, essentially a differential form of the epipolar constraint, building on earlier work by Heeger and Jepson [9].

Perhaps most relevant to the work presented in this paper, Gurfil and Rotstein [10] applied an IEKF with the subspace constraint for the estimation of aircraft states. They employed a simulated nonlinear aircraft model and used a set of tracked feature points to estimate the velocity, angular rates, angle of attack, and sideslip angle, starting from zero initial conditions. In these numerical studies, the full nonlinear equations of motion were used as the dynamic propagator in the Kalman filter. Because the subspace constraint allows for the estimation of velocity only up to a multiplicative constant, the velocity estimates were slow to converge to the simulated trajectory. However, the angle of attack and sideslip angle, which depend on the ratios of the velocity components, were estimated with a high degree of accuracy. Accurate results were also obtained for the angular rate estimates. In addition, Gurfil and Rotstein [10] also studied the performance of their algorithm in the presence of wind gust disturbances and measurement noise in the focal plane.

The approach taken in this paper complements the work of Gurfil and Rotstein [10], but also differs from their work in several respects. In this paper, the IEKF is employed using the epipolar constraint as opposed to the subspace constraint. Whereas Gurfil and Rotstein [10] employed the full nonlinear aircraft equations of motion for the dynamic propagator in the Kalman filter, this paper investigates the robustness of the filter with respect to modeling errors. This is an important consideration for MAVs, and aircraft in general, because the aerodynamic forces and moments acting on the vehicle are typically not known to a high degree of accuracy. A hierarchy of dynamic models is studied in this paper for implementation in the vision-based Kalman filter. These models include a random walk model, a linearized MAV model, and a nonlinear MAV model with simplified aerodynamics. In all cases, there is significant modeling error which must be accommodated by the vision-based Kalman filter. The performance of the vision-based state estimation algorithm is evaluated using simulated trajectories from a nonlinear MAV model developed at the NASA Langley Research Center [4]. The ability of the filter to cope with erroneous initial conditions is also studied as well as the robustness of the filter with respect to measurement noise in the focal plane.

Vehicle Equations of Motion and Observation Operators

In this section, we derive the coupling between the standard camera model and the equations of motion for the MAVs under consideration. The generic kinematics of a MAV in flight are depicted in Fig. 1. In this figure, the inertial, body-fixed, and k th camera reference frames are denoted \mathbf{E} , \mathbf{B} , and \mathbf{C}_k , respectively. A right-handed, orthonormal basis for each reference frame is denoted by lowercase unit vectors. For example, $\{\hat{b}_1, \hat{b}_2, \hat{b}_3\}$ constitutes the orthonormal basis for the body-fixed frame \mathbf{B} located at the center of mass of the aircraft.

The state of the aircraft is described in terms of the velocity \mathbf{v}_c of the center of mass expressed in the body-fixed frame, the angular velocity ${}^E\boldsymbol{\omega}^B$ of the body-fixed frame relative to the inertial frame, and the position \mathbf{R}_c of the center of mass of the aircraft in the inertial frame. In terms of components, these vectors are given by

$$\mathbf{v}_c = u\hat{b}_1 + v\hat{b}_2 + w\hat{b}_3 \quad {}^E\boldsymbol{\omega}^B = p\hat{b}_1 + q\hat{b}_2 + r\hat{b}_3$$

$$\mathbf{R}_c = X_{c,1}\hat{e}_1 + X_{c,2}\hat{e}_2 + X_{c,3}\hat{e}_3 \quad (1)$$

The orientation of the body-fixed frame relative to the inertial frame is determined in the usual way through the roll, pitch, and yaw angles (ϕ, θ, ψ) . In the most general case, the aircraft may be instrumented with multiple cameras. The position and orientation of the k th camera reference frame with respect to the body-fixed frame is depicted in Fig. 2. The geometry of the camera is determined by the roll, pitch, and yaw angles $(\phi_k, \theta_k, \psi_k)$ that orient the k th camera reference frame relative to the body frame. In addition, the location of the lens of the k th camera relative to the origin of the body-fixed frame (i.e., the vehicle center of mass) is given by the vector $\Delta_k^T = \{\Delta_{1,k}, \Delta_{2,k}, \Delta_{3,k}\}$. Thus, the parameters characterizing the k th camera can be assembled in vector form as

$$\boldsymbol{\alpha}_k^T(t) = \{\Delta_1, \Delta_2, \Delta_3, \phi, \theta, \psi, f\}_k$$

$$= \{\Delta_{1,k}, \Delta_{2,k}, \Delta_{3,k}, \phi_k, \theta_k, \psi_k, f_k\}$$

The parameter f_k is the focal length of the k th camera. Clearly, numerous other intrinsic camera parameters can be added to and considered in this vector including radial lens distortion, charge-coupled device (CCD) array misalignment, and others. These considerations are outside the scope of this paper, however.

As illustrated in Fig. 3, it is assumed that the image processing methodologies summarized in [11,12] have reduced the full camera

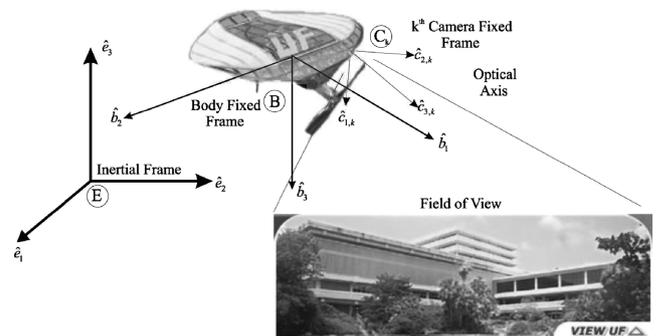


Fig. 1 Kinematics of the vehicle, camera, and field of view.

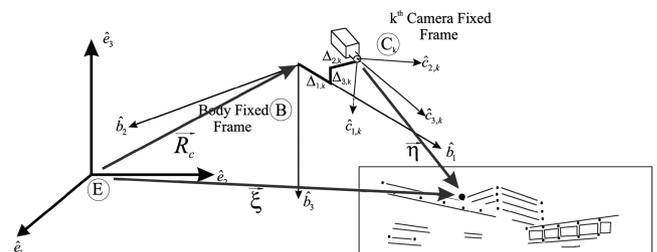


Fig. 2 Aircraft, camera, and feature point geometry.

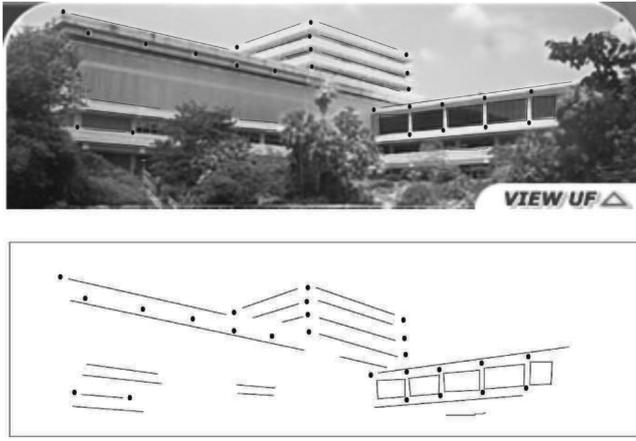


Fig. 3 Mapping of the camera view to feature points, lines, and planes.

image to obtain feature points, lines, and/or planes. In this paper, it is assumed that a family of feature points are processed and identified for guidance, navigation, and control. These feature points are assumed to be fixed in inertial space. Figure 2 illustrates the essential geometry between the aircraft body-fixed frame, the k th camera frame of reference, and the family of feature points obtained from the camera image. In this figure, $\{\mathbf{R}_c, \boldsymbol{\xi}, \boldsymbol{\eta}\}$ denote the location of the center of mass of the aircraft, the inertial location of the feature point, and the camera-relative location of the feature point.

The following expression can be derived for the camera-relative location of the p th feature point, expressed in the k th camera frame:

$$\boldsymbol{\eta}_p = R_{BC_k} R_{EB} \{\boldsymbol{\xi}_p - \mathbf{R}_c\} - R_{BC_k} \boldsymbol{\Delta}_k \quad (2)$$

In Eq. (2), R_{BC_k} denotes the orthogonal rotation matrix that defines the orientation of the k th camera reference frame \mathbf{C}_k relative to the body-fixed frame \mathbf{B} . This rotation matrix can be written in terms of three single-axis rotations as

$$R_{BC_k} = l(\phi_k, 1)l(\theta_k, 2)l(\psi_k, 3) \quad (3)$$

where

$$\begin{aligned} l(\phi_k, 1) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_k & \sin \phi_k \\ 0 & -\sin \phi_k & \cos \phi_k \end{bmatrix} \\ l(\theta_k, 2) &= \begin{bmatrix} \cos \theta_k & 0 & -\sin \theta_k \\ 0 & 1 & 0 \\ \sin \theta_k & 0 & \cos \theta_k \end{bmatrix} \\ l(\psi_k, 3) &= \begin{bmatrix} \cos \psi_k & \sin \psi_k & 0 \\ -\sin \psi_k & \cos \psi_k & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (4)$$

Similarly, R_{EB} denotes the orthogonal rotation matrix that defines the orientation of the body-fixed frame \mathbf{B} relative to the inertial frame \mathbf{E} :

$$R_{EB} = l(\phi, 1)l(\theta, 2)l(\psi, 3) \quad (5)$$

where the matrices $l(\phi, 1)$, $l(\theta, 2)$, and $l(\psi, 3)$ are defined in the same manner as in Eq. (4) and (ϕ, θ, ψ) are the Euler angles that specify the attitude of the vehicle.

This paper considers a simplified system that includes a single camera located at the aircraft center of mass. This camera is oriented at a fixed angle θ_1 relative to the body-fixed z axis. In other words, the transformation from body-fixed to camera axes consists of a single-axis rotation $l(\theta_1, 2)$ about the body-fixed y axis. In this case, Eq. (2) takes the simplified form

$$\boldsymbol{\eta}_p = l(\theta_1, 2)R_{EB} \{\boldsymbol{\xi}_p - \mathbf{R}_c\} \quad (6)$$

and the time derivative of the p th feature point position vector can be written as

$$\begin{aligned} \dot{\boldsymbol{\eta}}_p &= l(\theta_1, 2)[{}^E\boldsymbol{\omega}^B \times]R_{EB} \{\boldsymbol{\xi}_p - \mathbf{R}_c\} - l(\theta_1, 2)R_{EB} \dot{\mathbf{R}}_c \\ &= l(\theta_1, 2)[{}^E\boldsymbol{\omega}^B \times]l(\theta_1, 2)^T \boldsymbol{\eta}_p - l(\theta_1, 2)\mathbf{v}_c \end{aligned} \quad (7)$$

In Eq. (7), $[{}^E\boldsymbol{\omega}^B \times]$ is a skew symmetric matrix that computes the vector product:

$$[{}^E\boldsymbol{\omega}^B \times] = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \quad (8)$$

With the preceding definitions, it is possible to develop the equations that couple the aircraft equations of motion with the trajectories of tracked feature points in the focal plane. The collection of vehicle states are defined in the vector

$$\mathbf{X}^T(t) = \{u, v, w, p, q, r, \phi, \theta, \psi, X_{c,1}, X_{c,2}, X_{c,3}\}$$

The vector $\mathbf{U}(t)$ denotes the set of control inputs to the aircraft equations of motion and the family of trajectories of the n_{fp} feature points in the focal plane are defined as

$$\mathbf{Y}_p = f_k \begin{Bmatrix} \mu_p \\ \nu_p \\ 1 \end{Bmatrix} \in \mathbb{R}^3, \quad p = 1, \dots, n_{fp} \quad (9)$$

Finally, the camera parameters can be collected into a single vector $\boldsymbol{\alpha}^T = \{\boldsymbol{\alpha}_1^T, \dots, \boldsymbol{\alpha}_{n_{cam}}^T\}$, where n_{cam} is the number of cameras. In this paper, a perspective, or pinhole, projection camera model is employed. The perspective camera equations for the p th feature point are given by

$$\mu_p = \frac{\eta_{p,x}}{\eta_{p,z}} \quad \nu_p = \frac{\eta_{p,y}}{\eta_{p,z}} \quad (10)$$

where (μ_p, ν_p) denotes the position of the p th feature point in the focal plane.

Using the aircraft equations of motion, the above definitions, and the kinematic relations in Eqs. (6) and (10), the following set of equations is obtained, coupling the aircraft equations of motion and the measured trajectories of points in the image plane:

$$\begin{aligned} \dot{\mathbf{X}}(t) &= \mathbf{F}(\mathbf{X}(t), \mathbf{U}(t), t) & \mathbf{X}(0) &= \mathbf{X}_0 \\ \mathbf{Y}(t) &= \mathbf{C}(\mathbf{X}(t), \boldsymbol{\alpha}, \boldsymbol{\xi}(t), t) \end{aligned} \quad (11)$$

These equations will be referred to as the control theoretic form of the governing aircraft/camera equations. Alternatively, if Eq. (10) is differentiated in time, we obtain

$$\dot{\mu}_p = \frac{\dot{\eta}_{p,x}}{\eta_{p,z}} - \frac{\eta_{p,x}}{(\eta_{p,z})^2} \dot{\eta}_{p,z} \quad \dot{\nu}_p = \frac{\dot{\eta}_{p,y}}{\eta_{p,z}} - \frac{\eta_{p,y}}{(\eta_{p,z})^2} \dot{\eta}_{p,z} \quad (12)$$

If Eqs. (7) and (12) are employed instead of Eqs. (6) and (10), we obtain the alternative form

$$\begin{aligned} \dot{\mathbf{X}}(t) &= \mathbf{F}(\mathbf{X}(t), \mathbf{U}(t), t) & \mathbf{X}(0) &= \mathbf{X}_0 \\ \mathbf{Z}(t) &= \mathbf{G}(\mathbf{X}(t), \boldsymbol{\alpha}, \boldsymbol{\xi}(t), t) \end{aligned} \quad (13)$$

This latter set of equations will be referred to as the optic flow form of the governing aircraft/camera equations.

Kalman Filter Implementation

The estimation of aircraft states is achieved using an implementation of the extended Kalman filter [5]. The standard Kalman filter employs a linear dynamic model to propagate the states and the state covariance matrix. When considering nonlinear models, a common practice is to linearize the equations of motion about a nominal trajectory, or trim condition. The accuracy of such an approximation tends to deteriorate over time, however, as the actual trajectory often deviates substantially from the nominal one. The extended Kalman filter was developed as an alternative approach

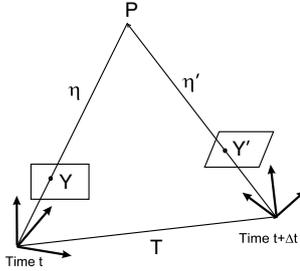


Fig. 4 The geometry of the epipolar constraint.

whereby the equations of motion are linearized about the current state estimate. Therefore, a linearized model is computed at each time step. The resulting linearized state transition matrix is then used for propagating the state covariance matrix. The original nonlinear equations of motion are typically used for state propagation. The standard implementation of the extended Kalman filter uses measurements that are explicit functions of the states. The approach taken in this paper is to use the positions of tracked feature points to update the state estimates, resulting in measurements that are implicit functions of the aircraft states, the tracked feature points, and the camera parameters. Soatto et al. [7] and Soatto and Perona [8] employed two distinct implicit measurement constraints that can be incorporated into the extended Kalman filter: the epipolar constraint and the subspace constraint. The resulting Kalman filtering algorithm has been termed the implicit extended Kalman filter.

Epipolar Constraint

The implicit measurement constraint employed in this paper is the epipolar constraint, also referred to in the literature as the essential or coplanarity constraint [13,14]. This constraint is formulated based on the principle that the position vectors of a static feature point relative to the camera at two instants in time must be coplanar with the translation vector of the origin of the camera frame. This geometric relationship is illustrated in Fig. 4, where η and η' denote the position vectors of feature point P in the camera reference frame, Y and Y' represent the position vectors projected onto the focal plane, and T indicates the translation vector of the camera. The epipolar constraint can be expressed as

$$\eta' \cdot (T \times R\eta) = 0 \quad (14)$$

or, equivalently, since the perspective projection camera model implies that η and η' are colinear with their projections Y and Y' onto the focal plane,

$$Y' \cdot (T \times RY) = 0 \quad (15)$$

Note that Y and Y' are three-dimensional vectors as defined in Eq. (9). Equations (14) and (15) reflect the fact that the scalar triple product of three coplanar vectors is zero. In these equations, R denotes the rotation matrix that describes the change in orientation of the camera from time t to time $t + \Delta t$. This transformation matrix can be expressed as

$$R = [l(\theta_1, 2)R_{EB}(\phi, \theta, \psi, t + \Delta t)][l(\theta_1, 2)R_{EB}(\phi, \theta, \psi, t)]^T \quad (16)$$

where $R_{EB}(\phi, \theta, \psi, t)$ and $R_{EB}(\phi, \theta, \psi, t + \Delta t)$ represent the rotations from inertial to body-fixed axes at times t and $t + \Delta t$, respectively. Recall that, for the examples considered in this paper, $l(\theta_1, 2)$ denotes the constant single-axis rotation from body-fixed axes to camera axes. Note that, in Eqs. (14) and (15), the translation vector T is expressed in the camera-fixed reference frame at time $t + \Delta t$.

Given the position vectors of n_{fp} feature points in the focal plane, as defined in Eq. (9), each must satisfy the epipolar constraint. This yields n_{fp} constraint equations:

$$[\mu'_p \ v'_p \ 1]E[\mu_p \ v_p \ 1]^T = 0, \quad p = 1, \dots, n_{fp} \quad (17)$$

where E denotes the essential matrix, defined as $[T \times]R$. By inspection, $[T \times]$ is the skew symmetric matrix that computes the vector product. Following the development of Soatto et al. [7], these constraints can be combined into a single equation

$$Ce = 0 \quad (18)$$

where the p th row of $C \in \mathbb{R}^{n_{fp} \times 9}$ takes the form

$$C_p = [\mu_p \mu'_p \ \mu_p v'_p \ \mu_p \ v_p \mu'_p \ v_p v'_p \ v_p \ \mu'_p \ v'_p \ 1]$$

and e denotes a vector composed of the stacked columns of E .

The measurement used in the IEKF at discrete-time t_k is given as

$$\hat{z}_k = C_{k-1,k} e_k \quad (19)$$

where $C_{k-1,k}$ denotes the matrix C composed of the feature point position measurements at times t_{k-1} and t_k . Note that the true value z_k of this measurement must be a vector of zeros because the epipolar constraint reflects a geometric relationship that is always satisfied in reality. The weighted difference between the measured value \hat{z}_k and the true value $z_k = \mathbf{0}$ is used in the measurement update in the Kalman filter.

Implicit Extended Kalman Filter

The extended Kalman filtering algorithm is well known; however, it is reviewed here to highlight the minor implementation differences that result from using measurements that are implicit functions of the aircraft states. The filter assumes that the state vector evolves according to a dynamic model as

$$X_k = F_{k-1}(X_{k-1}, U_{k-1}) + w_{k-1}, \quad k = 1, \dots, N \quad (20)$$

where N is the total number of estimated samples, and X_k and X_{k-1} denote the state at times t_k and t_{k-1} , respectively. Similarly, U_{k-1} represents the control input at time t_{k-1} . The Kalman filter assumes additive and uncorrelated zero-mean Gaussian process noise such that

$$\begin{aligned} w_k &= \mathcal{N}(0, Q_k) \quad \forall k \in [0, N-1] \\ E[w_i w_k^T] &= \delta_{ik} Q_k \quad \forall i, k \in [0, N-1] \end{aligned} \quad (21)$$

where E denotes the expected value and δ_{ik} is the Kronecker delta function. The entries of the matrix Q_k , which correspond to the covariance of the process noise, are tuning parameters in the filter.

A discrete-time implementation of the Kalman filter has been chosen because the measurements are available at discrete-time intervals. In the standard Kalman filter, the measurements are explicit functions of the states and the filter employs a measurement model of the form

$$Y_k = C_k(X_k) + v_k \quad (22)$$

where the measurement noise is also assumed to be additive, uncorrelated, zero mean, and Gaussian:

$$\begin{aligned} v_k &= \mathcal{N}(0, R_k) \quad \forall k \in [0, N-1] \\ E(v_i v_k^T) &= \delta_{ik} R_k \quad \forall i, k \in [0, N-1] \end{aligned} \quad (23)$$

In addition, the measurement noise is assumed to be uncorrelated with the process noise:

$$E(v_i w_k^T) = 0 \quad \forall i, k \in [0, N-1] \quad (24)$$

In the implicit extended Kalman filter employed in this paper, the measurement equation takes the form

$$z_k = h_k(X_k, Y_k, Y_{k-1}) = \mathbf{0} \quad (25)$$

The measurement in the filter, therefore, takes the form of a constraint that is an implicit function of the states and the measured feature point positions in consecutive image frames. It should be noted that the

feature point positions are the data that are physically measured via image processing. It is common in practice to assume that these measured feature point positions are corrupted by additive, zero-mean, Gaussian white noise.

The Kalman filter must first be initialized by choosing an initial estimate \hat{X}_0 of the state vector and an initial value P_0 for the state covariance matrix. The covariance matrices of the process noise Q_k and measurement noise R_k , which correspond to tuning parameters in the filter, must also be selected. In the implementation used in this paper, both Q_k and R_k are chosen to be diagonal matrices that remain constant for the duration of the estimation process. In setting the filter parameters, R_k corresponds to the noise in the measured feature point positions Y_k , not the noise in the implicit measurement \hat{z}_k .

The Kalman filter is a two-step procedure consisting of a propagation step using the dynamic model and a measurement update step. The state is first propagated as

$$\hat{X}_k^- = \mathbf{F}_{k-1}(\hat{X}_{k-1}^+, \mathbf{U}_{k-1}) \quad (26)$$

where the superscripts $-$ and $+$ denote the values of the state estimates after the propagation step and after the measurement update steps, respectively. In the implementation used in this paper, the state propagation in Eq. (26) is computed via fourth-order Runge–Kutta numerical integration of the continuous aircraft equations of motion. The state covariance matrix is also propagated using the equation

$$P_k^- = \Phi_{k-1} P_{k-1}^+ \Phi_{k-1}^T + Q_{k-1} \quad (27)$$

where Φ_{k-1} is obtained by linearizing the nonlinear equations of motion about the most recent state estimate:

$$\Phi_{k-1} = \left. \frac{\partial \mathbf{F}_{k-1}}{\partial \mathbf{X}} \right|_{\mathbf{X}=\hat{X}_{k-1}^+} \quad (28)$$

In the measurement update step, the implicit measurement \hat{z}_k is computed using the current state estimate \hat{X}_k^- and the tracked feature point positions Y_k and Y_{k-1} :

$$\hat{z}_k = \mathbf{h}_k(\hat{X}_k^-, Y_k, Y_{k-1}) = C_{k-1,k}(Y_k, Y_{k-1}, \boldsymbol{\alpha}) \mathbf{e}_k(\hat{X}_k^-, \boldsymbol{\alpha}) \quad (29)$$

Note that Eq. (29) is the same as Eq. (19) except that the dependence of the terms on the state estimate, feature point measurements, and camera parameters has been explicitly stated in Eq. (29). The implicit measurement is dependent on the measured positions of feature points that have been tracked in consecutive frames. Therefore, the vectors Y_{k-1} and Y_k are composed of the positions of only those points that have been tracked in the image plane at both times t_{k-1} and t_k . The measurement update to the state estimate then takes the form

$$\hat{X}_k^+ = \hat{X}_k^- + K_k(\mathbf{z}_k - \hat{z}_k) = \hat{X}_k^- - K_k \hat{z}_k \quad (30)$$

where K_k is the Kalman gain matrix and Eq. (30) makes use of the fact that the true value of the measurement constraint is zero: $\mathbf{z}_k = \mathbf{0}$. The state covariance matrix is also updated:

$$P_k^+ = [I - K_k H_k] P_k^- \quad (31)$$

In Eq. (31), H_k is obtained by linearizing the measurement equation about the state estimate \hat{X}_k^- :

$$H_k = \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{X}} \right|_{\mathbf{X}=\hat{X}_k^-} = C_{k-1,k} \left. \frac{\partial \mathbf{e}_k}{\partial \mathbf{X}} \right|_{\mathbf{X}=\hat{X}_k^-} \quad (32)$$

Before the state estimate and state covariance can be updated in Eqs. (30) and (31), the Kalman gain matrix K_k is computed as

$$K_k = P_k^- H_k^T [H_k P_k^- H_k^T + \tilde{R}_k]^{-1} \quad (33)$$

where \tilde{R}_k is the first-order approximation of the covariance of the noise in the implicit measurement constraint. This approximation is related to the covariance of the noise in the measured feature point

positions R_k as

$$\tilde{R}_k = D_k R_k D_k^T \quad (34)$$

where

$$D_k = \frac{\partial \mathbf{z}_k}{\partial \tilde{Y}_k} = \frac{\partial C_{k-1,k}}{\partial \tilde{Y}_k} \mathbf{e}_k \quad (35)$$

The vector \tilde{Y}_k is composed of elements of the feature point position measurements Y_{k-1} and Y_k :

$$\tilde{Y}_k = \{\mu_1, \nu_1, \mu'_1, \nu'_1, \dots, \mu_{n_{fp}}, \nu_{n_{fp}}, \mu'_{n_{fp}}, \nu'_{n_{fp}}\}^T$$

The primed variables represent feature point positions at time t_k and the unprimed variables represent feature point positions at time t_{k-1} .

It should be noted that, in general, this IEKF implementation incorporates several standard assumptions about the process and measurement noise that are not strictly satisfied in practice. The process noise, which must account for error in the dynamic propagator, generally does not adhere to the Gaussian white noise assumption. The degree to which the modeling error deviates from the process noise assumption is highly related to the aircraft trajectory. For example, the error in a linearized model about a wings-level trim condition will deviate substantially from Gaussian white noise as the vehicle performs a banked turn maneuver. Still, the Kalman filter is frequently applied under such circumstances and the inaccuracy of the process noise model represents one of several factors affecting the filter performance. Furthermore, although the noise in the measured feature point positions may be approximated as additive Gaussian white noise, this does not imply that the noise in the implicit measurement \mathbf{z}_k has the same characteristics. This noise is modeled in terms of the feature point measurement noise using the first-order approximation given in Eqs. (34) and (35). In addition, the implicit measurement is a function of measured feature point data in two consecutive image frames. Therefore, the measurement noise is clearly correlated within one measurement time step. The measurement noise is also typically correlated with the process noise, violating yet another assumption in the filter. Soatto et al. [7] investigated these issues in some detail. In particular, they showed that the feature point measurement noise can be decorrelated from itself and from the process noise by introducing additional noise states into the filter. This modification requires an additional state for each feature point measurement. Soatto et al. [7] concluded that, in most cases, the modest gain in performance does not justify the increase in filter complexity. Therefore, this modification was not explored in the research presented in this paper.

Dynamic Models

A number of dynamic models are considered in the filters employed in this paper. These models are based on the 12-state aircraft equations of motion which take the general form [15]:

$$\dot{v}_T = \frac{F_x}{m}$$

$$\dot{\beta} = p \sin \alpha - r \cos \alpha + \frac{F_y}{mv_T}$$

$$\dot{\alpha} = -p \tan \beta + q + \frac{F_z}{mv_T \cos \beta}$$

$$\dot{p} = \left[\frac{(I_{11} - I_{22} + I_{33})I_{13}}{I_{11}I_{33} - I_{13}^2} \right] p q + \left[\frac{(I_{22} - I_{33})I_{33} - I_{13}^2}{I_{11}I_{33} - I_{13}^2} \right] q r$$

$$+ \frac{I_{33}}{I_{11}I_{33} - I_{13}^2} \mathcal{L} + \frac{I_{13}}{I_{11}I_{33} - I_{13}^2} \mathcal{N}$$

$$\dot{q} = \left[\frac{I_{33} - I_{11}}{I_{22}} \right] p r - \frac{I_{13}}{I_{22}} (p^2 - r^2) + \frac{1}{I_{22}} \mathcal{M}$$

$$\dot{r} = \left[\frac{(I_{11} - I_{22})I_{11} + I_{13}^2}{I_{11}I_{33} - I_{13}^2} \right] pq - \left[\frac{(I_{11} - I_{22} + I_{33})I_{13}}{I_{11}I_{33} - I_{13}^2} \right] qr$$

$$+ \frac{I_{13}}{I_{11}I_{33} - I_{13}^2} \mathcal{L} + \frac{I_{11}}{I_{11}I_{33} - I_{13}^2} \mathcal{N}$$

$$\dot{\phi} = p + (q \sin \phi + r \cos \phi) \tan \theta$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (36)$$

$$\dot{\psi} = (q \sin \phi + r \cos \phi) \sec \theta$$

$$\dot{X}_{c,1} = v_T \cos \theta \cos \psi \cos \alpha \cos \beta + v_T (\sin \phi \sin \theta \cos \psi$$

$$- \cos \phi \sin \psi) \sin \beta + v_T (\cos \phi \sin \theta \cos \psi$$

$$+ \sin \phi \sin \psi) \sin \alpha \cos \beta$$

$$\dot{X}_{c,2} = v_T \cos \theta \sin \psi \cos \alpha \cos \beta + v_T (\sin \phi \sin \theta \sin \psi$$

$$+ \cos \phi \cos \psi) \sin \beta + v_T (\cos \phi \sin \theta \sin \psi$$

$$+ \sin \phi \cos \psi) \sin \alpha \cos \beta$$

$$\dot{X}_{c,3} = -v_T \sin \theta \cos \alpha \cos \beta + v_T \sin \phi \cos \theta \sin \beta$$

$$+ v_T \cos \phi \cos \theta \sin \alpha \cos \beta$$

The aircraft state vector includes three velocity states $\{v_T, \beta, \alpha\}$, three angular velocity states $\{p, q, r\}$, three Euler angles $\{\phi, \theta, \psi\}$, and three inertial position states $\{X_{c,1}, X_{c,2}, X_{c,3}\}$. The velocity states are given in terms of wind axes (i.e., the relative wind is oriented in the x direction) and correspond to total velocity, sideslip angle, and angle of attack. This is an alternative representation of the aircraft velocity to the body-fixed velocity vector $\mathbf{v}_c = \{u, v, w\}^T$. These representations are related as

$$u = v_T \cos \alpha \cos \beta \quad v = v_T \sin \beta \quad w = v_T \sin \alpha \cos \beta \quad (37)$$

The attitude is expressed in terms of the Euler angles, which correspond to the standard roll, pitch, and yaw angles $\{\phi, \theta, \psi\}$. The inertial position states correspond to a north, east, down inertial coordinate system. In Eq. (36), the I_{ij} terms are elements of the inertia matrix about the center of mass. The terms $\{F_x, F_y, F_z\}$ correspond to the forces acting on the aircraft, expressed in wind axes:

$$F_x = -D \cos \beta + (T - W \sin \theta) \cos \alpha \cos \beta$$

$$+ W \cos \theta \cos \phi \sin \alpha \cos \beta + Y \sin \beta + W \cos \theta \cos \phi \sin \beta$$

$$F_y = D \sin \beta + (T - W \sin \theta) \cos \alpha \sin \beta$$

$$- W \cos \theta \cos \phi \sin \alpha \sin \beta + Y \cos \beta + W \cos \theta \cos \phi \cos \beta$$

$$F_z = -L - (T - W \sin \theta) \sin \alpha + W \cos \theta \cos \phi \cos \alpha \quad (38)$$

where $\{W, L, D, T, Y\}$ correspond to the weight, lift, drag, thrust, and side force. The terms $\{\mathcal{L}, \mathcal{M}, \mathcal{N}\}$ in Eq. (38) are the aerodynamic moments acting on the aircraft.

Although the general form of the aircraft equations of motion is well known, it is extremely difficult to obtain an accurate model of the aerodynamic forces $\{L, D, Y\}$ and moments $\{\mathcal{L}, \mathcal{M}, \mathcal{N}\}$, especially for small aircraft such as MAVs that operate in low

Reynolds number regimes. Therefore, there will always be some level of uncertainty in the aircraft model. The aerodynamic forces and moments are nonlinearly dependent on the states, and the form of this dependency is unique to the specific aircraft being studied. The specific model considered in this paper is a nonlinear MAV model developed at the NASA Langley Research Center [4].

In this paper, a hierarchy of models is considered for use in the vision-based Kalman filter. These models assume varying levels of knowledge about the dynamics of the system. The filter estimates eight states: three velocity states, three angular velocity states, the roll angle ϕ , and the pitch angle θ . The other four states are not directly observable from vision and are therefore not included in the filter. The first model considered propagates the velocity and angular velocity states by adding zero-mean Gaussian white noise. This corresponds to a first-order random walk model for these states and fits in well in the framework of the Kalman filter, which assumes Gaussian white process noise. Note that the equations of motion for the roll and pitch angles are known exactly from kinematics. Therefore, those equations are included in the dynamic model, as suggested by Chiuso et al. [16]. The second model corresponds to a linear model obtained by linearizing the MAV equations of motion about an equilibrium, or trim, condition. The accuracy of this model, of course, decreases as the aircraft performs maneuvers away from the trim condition. The final model corresponds to a nonlinear model that retains all of the known nonlinear terms due to inertial coupling but employs simplified aerodynamics. An important consideration in this paper is to evaluate the performance of the vision-based state estimator using these different aircraft models.

Numerical Results for a Simulated Micro Air Vehicle

In this section, the performance of the vision-based state estimation algorithm is examined for a simulated MAV. The vehicle is a 6 in. MAV that was designed and built at the University of Florida. The MAV weighs about 0.12 lbs and has a maximum speed of approximately 50 ft/s. The control inputs for this vehicle correspond to a voltage input to the motor for thrust, symmetric elevon deflection for longitudinal motion, and antisymmetric elevon deflection for lateral motion. An open-loop simulation of the nonlinear equations of motion for this MAV was developed at the NASA Langley Research Center [4] and has been used to generate simulated MAV trajectories with which to test the estimation algorithm. In constructing this MAV model, Waszak et al. [4] collected detailed aerodynamic data at three values of dynamic pressure corresponding to $\{1.0, 1.6, 2.0\}$ psf at sea level. The aerodynamic coefficients were modeled in some cases using wind tunnel test data and in others using the software PMARC. The aerodynamic data in the nonlinear model are given via bilinear interpolation of the data obtained at the three test points. For further details regarding the Langley simulation model, please refer to [3,4].

The nonlinear MAV simulation model has been used to generate realistic MAV trajectories for evaluating the performance of the vision-based state estimation algorithm. It was also necessary to simulate the identification and tracking of feature points as viewed by a camera mounted at the aircraft center of mass. In these studies, the camera was oriented at an angle of 60 deg downward from the nose of the vehicle. The focal length of the simulated camera was selected to be 1 and the image plane was simulated to be 640 horizontal pixels and 480 vertical pixels, for a 4/3 aspect ratio. These camera properties correspond to a horizontal field of view of 60 deg and a vertical field of view of about 47 deg. The feature points were simulated by creating a cloud of randomly distributed points in inertial space. At each time during the simulated MAV trajectory, the positions of the feature points in the image plane were computed using the pinhole camera projection model. Those points that fell outside the camera field of view at any time were discarded for that particular frame. Therefore, the number of tracked feature points varied throughout the simulations, with points disappearing from the field of view and new points appearing. The feature point simulation matches corresponding points in consecutive frames, a task that would be performed by the feature point tracker in practice.

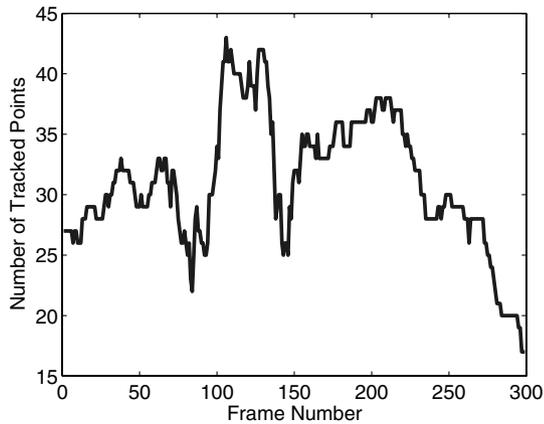


Fig. 5 Number of tracked feature points in each image frame during the simulated flight.

The vision-based state estimator was tested for a relatively simple trajectory consisting of a smoothed roll doublet maneuver in which the MAV rolled to the right roughly 30 deg, back to the left to a roll angle of -30 deg, and finally back to wings level. This trajectory, which was 30 s in duration, was generated via open-loop control commands in the MAV simulation. The vision-based measurements were simulated at a rate of 10 Hz so that tracked feature points were available 10 times per second. This is a reasonable assumption because digital video is typically acquired at 30 frames per second and, based on our personal experience with the Lucas–Kanade algorithm [11,12], feature point trackers can operate in real time. In Fig. 5, the number of tracked feature points is plotted as a function of the frame number. In total, there were 301 simulated frames, and the number of tracked feature points varied in the range of 17 to 43 points. In these numerical experiments, eight MAV states were estimated: the total velocity v_T , sideslip angle β , angle of attack α , the three components of angular velocity $\{p, q, r\}$, the roll angle ϕ , and the pitch angle θ .

As mentioned earlier, three dynamic models of varying fidelity were employed in the Kalman filter. The first model considered was a first-order random walk model for the velocity and angular velocity states with the exact kinematic expressions for the roll and pitch angles. The next model considered was a linear MAV model obtained through a linearization of the equations of motion about a trim condition corresponding to the initial conditions for the simulated flight. During the roll doublet, the simulated nonlinear trajectory deviates substantially from that predicted by the linearized model. After this maneuver, the MAV settles to a new trim condition different from the one used to generate the linearized model. Finally, a nonlinear MAV model was considered that contains simplified nonlinear aerodynamics. The aerodynamics in this model differ from those in the simulation in that the aerodynamic coefficients are assumed to have constant values at a dynamic pressure of 1.6 psf instead of interpolating between values at 1.0, 1.6, and 2.0 psf as in the original Langley MAV model. During the simulated flight, the MAV is initially operating at a dynamic pressure close to 1.6 psf but is at a dynamic pressure of approximately 1.2 psf at the end of the flight. In addition, the aerodynamics in the simplified nonlinear model do not include dynamic terms that depend on the time derivatives of the angle of attack and sideslip angle.

The random walk, linear, and nonlinear MAV models were employed in the vision-based Kalman filter to evaluate the robustness of the estimator with respect to modeling error. It should be noted that there are many other options for studying model robustness. For example, one could conduct a parametric study to determine the effect of errors in some of the individual model coefficients. This particular approach was not pursued in this paper, however. The linear and nonlinear models employed in this paper effectively contain errors in many of the coefficients, the amount of which varies throughout the simulated flight. Therefore, the robustness study presented in this paper represents one means for

evaluating the filter performance using inaccurate dynamic propagators.

In this numerical study, the effect of measurement noise was also considered by adding normally distributed white noise to each feature point position in the pixel plane. The standard deviation of the measurement noise was varied from 0 to 5 pixels while it is noted that 1 pixel error is typical of many image processing algorithms [7,17]. In each case, the filter was given initial conditions that were significantly in error. Specifically, the true initial condition $\{v_T, \beta, \alpha, p, q, r, \phi, \theta\}_0$ corresponded to

$$\begin{aligned} X_0 = \{ & 37 \text{ ft/s}, 0.034691 \text{ deg}, 5.4628 \text{ deg}, 0 \text{ deg/s}, \\ & 0 \text{ deg/s}, 0 \text{ deg/s}, -1.6451 \text{ deg}, 5.4628 \text{ deg} \} \end{aligned}$$

while the Kalman filter was initialized to

$$\begin{aligned} \hat{X}_0 = \{ & 27 \text{ ft/s}, 28.65 \text{ deg}, 28.65 \text{ deg}, 57.3 \text{ deg/s}, \\ & -57.3 \text{ deg/s}, 28.65 \text{ deg/s}, 57.3 \text{ deg}, 28.65 \text{ deg} \} \end{aligned}$$

Therefore, the filter was given significant initialization errors in all of the states. These erroneous initial conditions represent an important test for the filter since it is quite possible that accurate initial conditions will not be known in practice.

The performance of the vision-based Kalman filter using the different dynamic models is summarized in Figs. 6–14. These figures present plots of the estimated states for the duration of the flight. In this set of results, the error in the measured feature point positions was simulated to have a standard deviation of 1 pixel. Roughly speaking, during each simulation the estimator goes through two phases: a transient phase in which the estimator responds to initialization error and a steady-state phase in which the estimator attempts to track the state as the MAV maneuvers. Therefore, for clarity of presentation, the results are plotted for two different time intervals: a 0–5 s transient interval and a 5–30 s steady-state interval. In the plots, the estimated states are shown as solid lines and labeled in the legend as “estimated” while the true states, as simulated using the nonlinear Langley MAV model, are depicted as dashed lines and labeled “simulated.” Also included in these plots is the propagated response, or the response predicted by the dynamic model without the benefit of vision-based measurements. These responses are shown as dash-dotted lines and are labeled “propagated” in the legend. We note that estimation error plots, commonly used to evaluate the performance of Kalman filters, have not been included here due to space limitations. We have chosen to present plots of the estimated states instead of the estimation error plots so that the reader can better understand the nature of the simulated flight.

From Fig. 7, the random walk model performs well in estimating the angular velocity states. The estimator quickly recovers from the significant initialization error and tracks the states throughout the roll maneuver. The estimated roll and pitch states are not accurate, however, as the estimates never fully recover from the angle initialization errors. The estimated velocity states are essentially unaffected by the vision-based measurements and remain constant

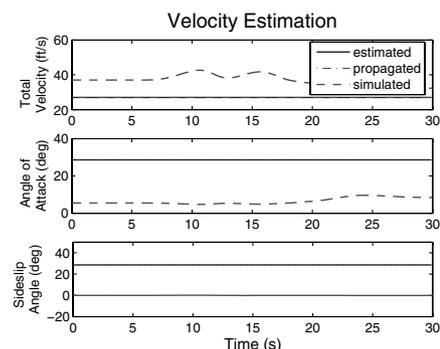


Fig. 6 Velocity estimation using the random walk dynamic model.

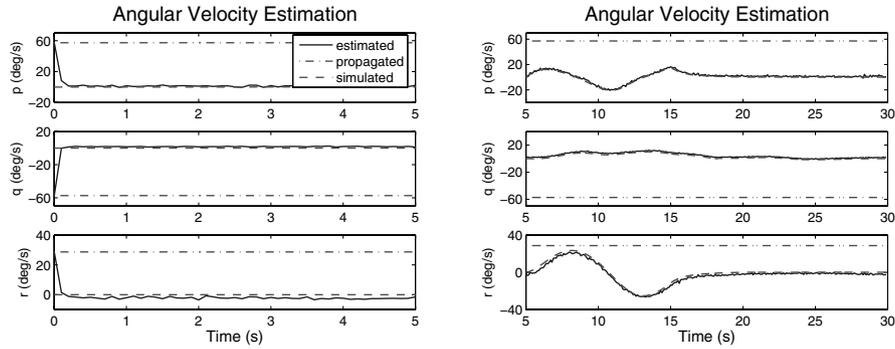


Fig. 7 Angular velocity estimation using the random walk dynamic model.

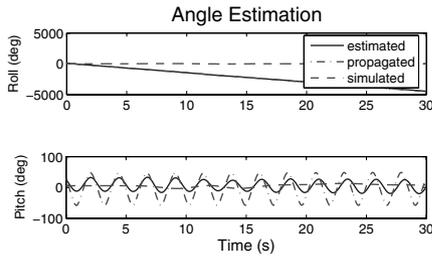


Fig. 8 Roll and pitch angle estimation using the random walk dynamic model.

for the entire simulation. It should be noted that there is a scale ambiguity in the epipolar constraint. Namely, any scalar multiple of the essential matrix satisfies the epipolar constraint. The result of this scale ambiguity is that, based on vision alone, the velocity can be determined only up to a scale factor. The use of a physical model in the Kalman filter provides information which can resolve the scale ambiguity. The random walk model is not physical and thus provides no scale information; therefore, it cannot estimate the total velocity. The random walk dynamic model alone is certainly insufficient for predicting the response of any of the states because it predicts

constant values for the angular velocity and velocity states. Although the model uses the exact kinematic expressions for the roll and pitch angles, on its own the model cannot compensate for the angle initialization errors. Therefore, the propagated roll and pitch angles are grossly in error.

The results for the estimator with the linear model are considerably different from those of the random walk model. Most notably, the linear model is stable and is able to recover from initial condition errors. As seen in Figs. 9–11, most of the propagated states eventually converge to the true simulated states. The estimated states, however, recover from the filter initialization errors much more rapidly in almost all cases. For example, in Fig. 10, the estimated angular velocity converges to the true simulated angular velocity in less than a quarter of a second. The propagated angular velocity also converges to the simulated angular velocity but requires about 2 s to do so. In addition, the propagated angular velocity states exhibit considerable oscillation before converging. The estimator also tracks the true angular velocity more accurately than the propagator throughout the simulated flight. This can best be seen in the plot of the estimated q component of angular velocity. As shown in Fig. 11, the estimator performs better than the propagator in estimating the roll and pitch angles. The estimator recovers from initial condition errors much more quickly, especially for the pitch angle, and tracks the roll and pitch states more closely during the entire simulation.

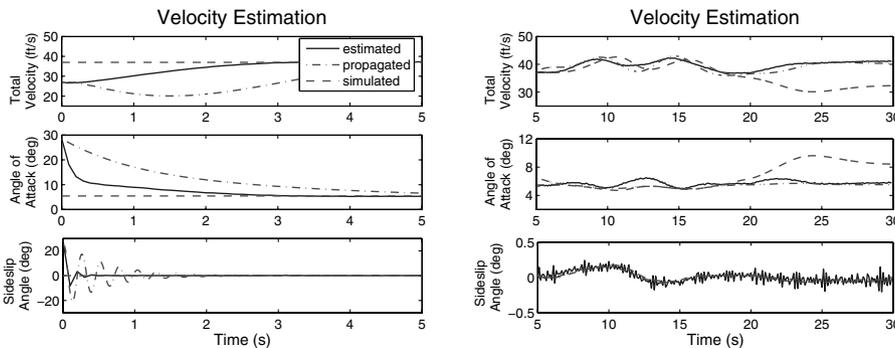


Fig. 9 Velocity estimation using the linear dynamic model.

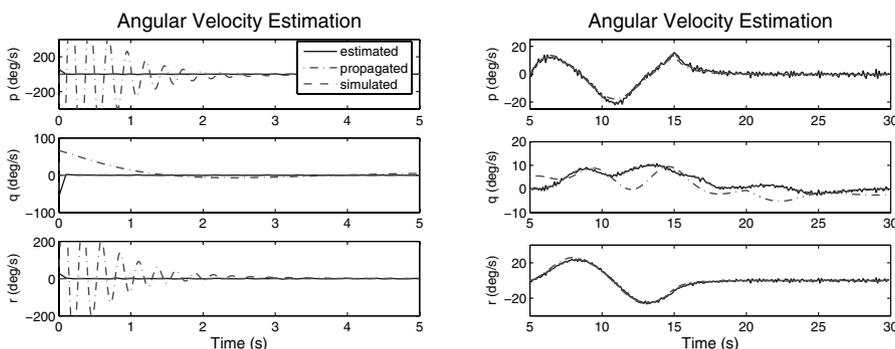


Fig. 10 Angular velocity estimation using the linear dynamic model.

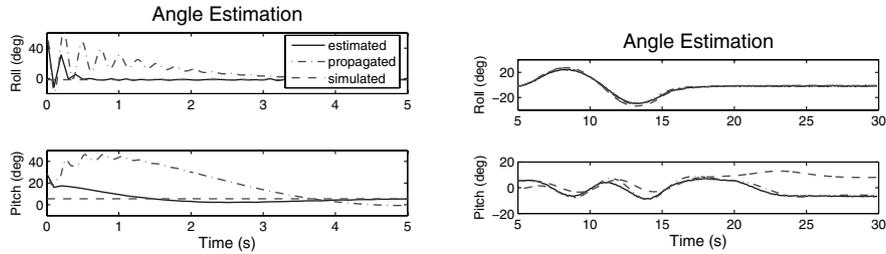


Fig. 11 Angle estimation using the linear dynamic model.

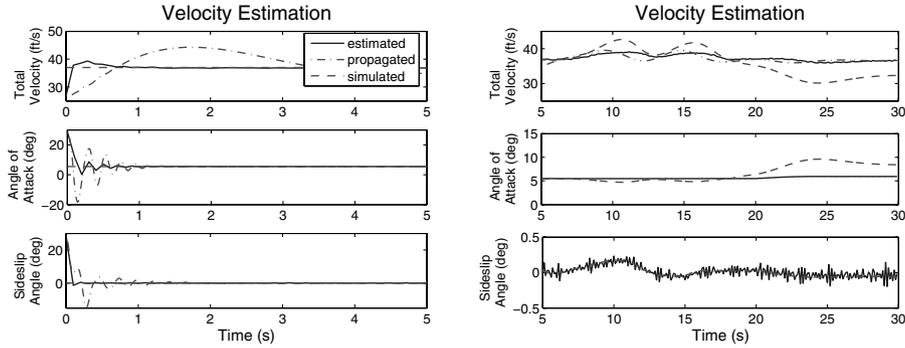


Fig. 12 Velocity estimation using the nonlinear dynamic model.

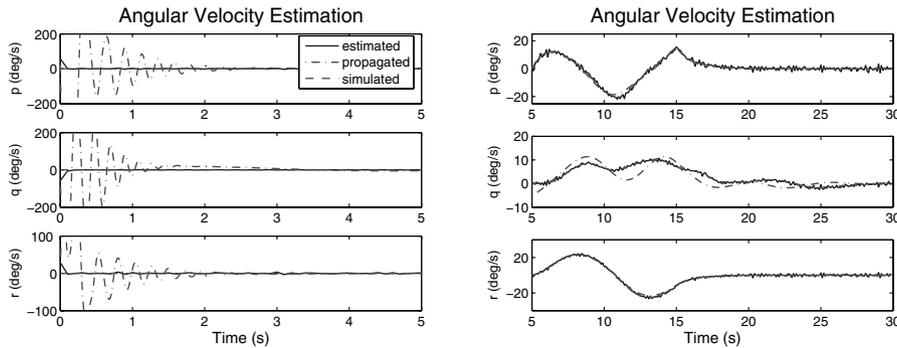


Fig. 13 Angular velocity estimation using the nonlinear dynamic model.

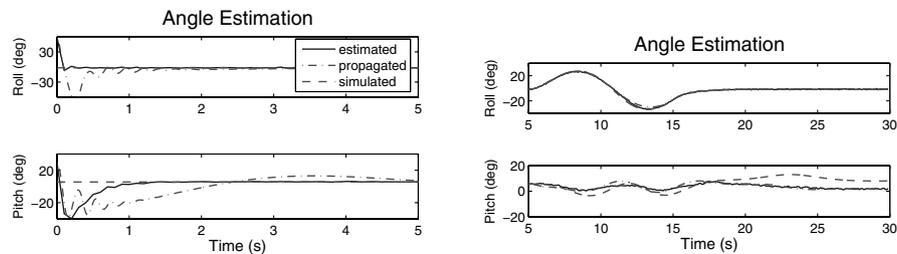


Fig. 14 Angle estimation using the nonlinear dynamic model.

The plot of the pitch angle shows that both the estimator and propagator deviate substantially from the true pitch angle toward the end of the simulation. This behavior is most likely due to the effect of errors in the estimation of the angle of attack. It is important to note that, after the roll maneuver is completed, the MAV settles to a new trim condition, significantly different from that used to generate the linearized dynamic model. As shown in Fig. 9, both the estimator and the propagator fail to track the resultant change in angle of attack. The angle of attack estimation error likely had an effect on the Kalman gain, which in turn affected the pitch angle estimation. In general, the estimator does not perform well in estimating the velocity states. It does, however, compensate for initial condition errors in the velocity states more quickly than the propagator. After recovering from the

initialization errors, the estimator does not track the velocity states very well in the long term for reasons that will be discussed later.

The state estimates obtained using the Kalman filter with the simplified nonlinear model are shown in Figs. 12–14. Qualitatively, the results are similar to those obtained using the linearized dynamic model. Similar to the linear model, the estimator performs very well in estimating the angular velocity states. The only difference is that the nonlinear propagator is slightly more accurate than the linear propagator, but the estimator performs similarly in both cases. It should be noted that the estimator also performs similarly when estimating the angular velocity using the random walk model. This result implies that the choice of dynamic model is not critical when estimating the angular velocity. For the nonlinear model, the roll and

pitch angle estimation is similar to that obtained with the linear model. The most significant difference is seen in the pitch angle estimation toward the end of the simulation. For similar reasons as in the linear case, the estimator with the nonlinear model shows significant error in the pitch angle estimation at the end of the simulation. Because of the higher fidelity of the nonlinear model, this error is much smaller in the nonlinear case. As in the linear case, the nonlinear estimator does not track the velocity states very well in the long term. This is especially evident in the estimation of the angle of attack shown in Fig. 12. An important difference is that the nonlinear estimator recovers from initial condition errors much faster than the linear estimator in estimating the total velocity and the angle of attack.

A prevailing observation from the results of testing the vision-based state estimator using the three different dynamic models is that, in all cases, the estimator is not able to track the velocity states well in the long term. The reason for this behavior is that the epipolar constraint has a stronger dependence on the angular velocity states than it does on the velocity states. Therefore, the partial derivatives of the implicit measurement with respect to the velocity states, which appear in the linearized measurement matrix H , are very small (see [18]). These small values lead to large gains since H is inverted in computing the Kalman gain matrix K , which in turn causes large biases in the state estimates. To alleviate this problem, it was necessary to set the partial derivatives of the measurement with respect to the velocity states to zero. This heuristic adjustment serves to eliminate the large biases in the state estimates, but also weakens the effect of the filter on estimating the velocity states. The filter is still able to provide measurement updates to the estimated velocity because, when the Kalman gain matrix is updated, the nonzero partial derivatives in H generally result in nonzero gains for all the states. Still, the results show that the vision-based measurement updates in the filter have a limited effect on the velocity estimation under these conditions.

Finally, the performance of the vision-based Kalman filter for varying levels of measurement noise is summarized in Fig. 15. This study was conducted using the estimator with the nonlinear dynamic model and the same simulated MAV flight and initial conditions used previously. The simulated feature point positions in the pixel plane were corrupted with additive and uncorrelated zero-mean Gaussian noise with standard deviation ranging from 0 to 5 pixels. Figure 15 depicts the rms estimation error in each estimated state for the varying noise levels. For each individual state, generically denoted as x , the rms estimation error is computed as:

$$e_{\text{rms}} = \left\{ \frac{1}{N+1} \sum_{k=0}^N (\hat{x}_k - x_k)^2 \right\}^{\frac{1}{2}} \quad (39)$$

where x_k denotes the true state at time t_k , \hat{x}_k represents the estimated state at time t_k , and $N + 1$ is the total number of estimated samples. As would be expected, the rms estimation error, shown as a solid line, is seen to gradually increase with the level of measurement noise for most of the states. For comparison, the rms error of the nonlinear

propagator without vision-based updates is shown as a dash-dotted line in each plot. This error is not dependent on the level of measurement noise because the measurements are not used in this case. The rms error in the estimated velocity states is similar for the estimator and the propagator. Once again, this is due to the vision-based filter not having a strong effect on the velocity estimates. The angular velocity states show an rms estimation error that increases gradually with the level of noise. The rms estimation error remains significantly smaller than the propagation error even as the measurement noise is increased to 5 pixels standard deviation. Note that the large rms error in the propagated angular velocity is due in large part to the large oscillatory response resulting from the initialization errors. Similarly, the rms error in the roll and pitch angle estimates remains smaller than the rms propagation error for all noise levels considered.

Conclusions

This paper has investigated the performance of a vision-based Kalman filter for state estimation for MAVs. The estimation is performed via an implementation of the IEKF that is based on the epipolar constraint. This constraint provides measurements that are implicit functions of the aircraft states and the measured feature point positions in the pixel plane. Numerical results have been presented for a simulated MAV performing a roll doublet maneuver. In these experiments, a nonlinear MAV model developed at the NASA Langley Research Center was used to generate the MAV trajectory. In this paper, a hierarchy of dynamic models of varying fidelity was employed to test the robustness of the algorithm with respect to the modeling error. These models included a random walk model, a linearized dynamic model, and a nonlinear model with simplified aerodynamics. The performance of the filter with respect to initialization errors and measurement noise in the pixel plane was also studied.

The simulation results indicate that the vision-based Kalman filter is capable of accurately estimating the angular velocity regardless of the dynamic model employed. Using all three models, the filter is able to account for errors in the initial conditions and track the angular velocity throughout the simulation. These results demonstrate that, in a qualitative sense, the angular velocity is highly observable from the epipolar constraint. Although a simple random walk model is sufficient for angular velocity estimation, the use of a higher-fidelity dynamic model is needed to estimate the velocity states and the roll and pitch angles. Similar results are obtained for the roll and pitch estimates using the linear and nonlinear models in the filter. The main difference is that the nonlinear model provides more accurate pitch estimates in the long term. The filter does not perform well in estimating the velocity states using either the linear or the nonlinear models. The epipolar constraint is weakly dependent on the velocity states, leading to small partial derivatives in the linearized measurement. To prevent biases, it was necessary to set these partial derivatives to zero, further weakening the ability of the filter to affect the velocity estimation. Under these conditions, the dynamic model becomes very important as it is the primary means by

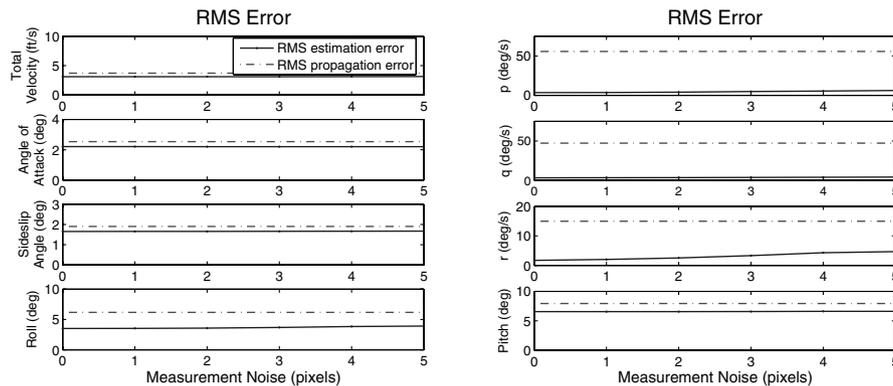


Fig. 15 RMS estimation errors for varying levels of noise in the measured feature point positions.

which the estimator can track the true velocity. The estimated velocity using the nonlinear model recovers much more rapidly from initialization errors than the velocity estimated using the linear model. The use of a high-fidelity dynamic model is also important in the event that the vision-based measurements become unavailable for a period of time. In practice, this could occur for a variety of reasons such as data dropouts, corrupted image frames, and flying over areas without trackable feature points. In these cases, the dynamic model would serve to propagate the state, albeit with increasing error, until another measurement became available. It is also important to note that most algorithms that estimate vehicle motion using vision alone, such as the well-known eight points algorithm, require at least eight tracked feature points to compute an estimate. Because the Kalman filter incorporates a dynamic model, it will produce state estimates regardless of the number of available feature points.

The results presented in this paper suggest many avenues for future work. The performance of the filter needs to be studied using real imagery from an urban environment and feature points obtained using a feature point tracker. In addition, the use of multiple cameras should be investigated. The results presented in this paper indicate that the lateral motion can be effectively estimated using a forward-looking camera whereas the longitudinal states are more difficult to estimate. This observation suggests that the use of a sideways-pointing camera might be more effective for estimating the longitudinal motion. In addition, the filter should be modified to incorporate data from any other sensors, such as GPS, accelerometers, or gyros, that may be included in a MAV's sensor suite. In principle, this should simply require appending additional measurements to the filter, although these measurements would be available on different time scales (e.g., GPS is available on the order of 1 Hz). Including other sensor measurements in the filter also provides a potential avenue for achieving full 12-state estimation. As noted earlier, the Kalman filter implementation used in this paper employs several assumptions about the process and measurement noise that are not satisfied in practice. This implies that improved results might be obtained using an unscented Kalman filter or a particle filter which makes no assumptions on the form of the process and measurement noise. Another possibility for future work includes using the subspace constraint, which is based on optic flow, in the filter instead of or in combination with the epipolar constraint. Finally, the ultimate research goal is to incorporate vision-based state estimation into an autonomous flight control system for MAVs.

Acknowledgments

The work presented in this paper was supported jointly by the Air Force Research Laboratory and the Air Force Office of Scientific Research under F49620-03-1-0381 with Johnny Evers, Neal Glassman, Sharon Heise, and Robert Sierakowski as project monitors. The authors would like to express their appreciation to Martin Waszak of the NASA Langley Research Center for providing the nonlinear MAV simulation model used to generate the numerical results in this paper. The authors would also like to thank the

anonymous reviewers for many useful suggestions that improved the quality of the paper.

References

- [1] Ettinger, S., Nechyba, M., Ifju, P., and Waszak, M., "Vision Guided Flight Stability and Control for Micro Air Vehicles," *IEEE International Conference on Intelligent Robots and Systems*, IEEE, Piscataway, NJ, Oct. 2002, Vol. 3, pp. 2134–2140.
- [2] Ifju, P., Ettinger, S., Jenkins, D., Lian, Y., Shyy, W., and Waszak, M., "Flexible-Wing-Based Micro Air Vehicles," *AIAA Aerospace Sciences Meeting and Exhibit*, AIAA, Reston, VA, 2002.
- [3] Waszak, M., Jenkins, L., and Ifju, P., "Stability and Control Properties of an Aeroelastic Fixed Wing Micro Air Vehicle," AIAA Paper 2001-4005, 2001.
- [4] Waszak, M., Davidson, J., and Ifju, P., "Simulation and Flight Control of an Aeroelastic Fixed Wing Micro Air Vehicle," AIAA Paper 2002-4875, 2002.
- [5] Grewal, M., and Andrews, A., *Kalman Filtering: Theory and Practice*, Prentice-Hall, Upper Saddle River, NJ, 1993.
- [6] Rogers, R., *Applied Mathematics in Integrated Navigation Systems*, AIAA, Reston, VA, 2000.
- [7] Soatto, S., Frezza, R., and Perona, P., "Motion Estimation via Dynamic Vision," *IEEE Transactions on Automatic Control*, Vol. 41, No. 3, March 1996, pp. 393–413.
- [8] Soatto, S., and Perona, P., "Recursive 3-D Motion Estimation Using Subspace Constraints," *International Journal of Computer Vision*, Vol. 22, No. 3, 1997, pp. 235–259.
- [9] Heeger, D., and Jepson, A., "Subspace Methods for Recovering Rigid Motion I: Algorithm and Implementation," *International Journal of Computer Vision*, Vol. 7, No. 2, 1992, pp. 95–117.
- [10] Gurfil, P., and Rotstein, H., "Partial Aircraft State Estimation from Visual Motion Using the Subspace Constraints Approach," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 5, Sept.–Oct. 2001, pp. 1016–1028.
- [11] Lucas, B., and Kanade, T., "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, William Kaufmann, Los Altos, CA, Aug. 1981, pp. 674–679.
- [12] Tomasi, C., and Kanade, T., "Shape and Motion from Image Streams: A Factorization Method—Part 3: Detection and Tracking of Point Features," Robotics Institute, Carnegie Mellon University, TR CMU-CS-91-132.
- [13] Luong, Q.-T., and Faugeras, O., "The Fundamental Matrix: Theory, Algorithms, and Stability Analysis," *International Journal of Computer Vision*, Vol. 17, No. 1, 1996, pp. 43–75.
- [14] Ma, Y., Soatto, S., Kosecka, J., and Sastry, S., *An Invitation to 3-D Vision*, Springer-Verlag, New York, 2004.
- [15] Etkin, B., and Reid, L., *Dynamics of Flight: Stability and Control*, Wiley, New York, 1996.
- [16] Chiuso, A., Favaro, P., Jin, H., and Soatto, S., "Motion and Structure Causally Integrated Over Time," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 4, 2002, pp. 523–535.
- [17] Barron, J., Fleet, D., Beauchemin, S., and Burkitt, T., "Performance of Optical Flow Techniques," *Proceedings of the 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Piscataway, NJ, June 1992.
- [18] Webb, T., Prazenica, R., Kurdila, A., and Lind, R., "Vision-Based State Estimation for Uninhabited Aerial Vehicles," AIAA Paper 2005-5869, 2005.