2010

Autonomous Feature Following for Visual Surveillance Using a Small Unmanned Aerial Vehicle with Gimbaled Camera System

International Journal of Control, Automation, and Systems (2010) 8(5):957-966 http://www.springer.com/12555
DOI 10.1007/s12555-010-0504-1

Autonomous Feature Following for Visual Surveillance Using a Small Unmanned Aerial Vehicle with Gimbaled Camera System

Deok-Jin Lee, Isaac Kaminer, Vladimir Dobrokhodov, and Kevin Jones

Abstract: This paper represents the development of feature following control and distributed navigation algorithms for visual surveillance using a small unmanned aerial vehicle equipped with a low-cost imaging sensor unit. An efficient map-based feature generation and following control algorithm is developed to make an onboard imaging sensor to track a target. An efficient navigation system is also designed for real-time position and velocity estimates of the unmanned aircraft, which is used as inputs for the path following controller. The performance of the proposed autonomous path following capability with a stabilized gimbaled camera onboard a small unmanned aerial robot is demonstrated through flight tests with application to target tracking for real-time visual surveillance.

Keywords: Autonomous navigation, imaging sensors, path following control, real-time visual surveillance, stabilized gimbaled camera, unmanned aerial robots.

1. INTRODUCTION

Unmanned aerial vehicles (UAVs) with onboard imaging and visual sensors play a strategic role in a broad range of applications, including visual surveillance, border control, and search missions [1]. Autonomy is a key technology for future high-performance of UAVs in various applications and challenging tasks for autonomy of the UAVs for successful real-time visual surveillance operations include autonomous flight path planning, localization and navigation, and onboard sensing systems [2,3].

Autonomous feature or path following capability is a logical extension of the emerging view of UAV as a flying sensor, combining wireless sensor network, advanced flight control algorithm, and the sensory capability of a camera onboard the unmanned aerial system (UAS) [1,3]. The feature following capability becomes extremely useful for high-level security and safety support tasks including visual surveillance and reconnaissance with real-time video and high-resolution image information.

The proposed concept of operation assumes that the surveilance mission is initiated remotely by a minimally trained user who only provides a sensor path on the ground by scribbling it on a digital map. This produces a precise continuous footprint for the airborne sensors to follow. The corresponding UAV flight path is optimized taking into account the resolution requirements, mission limitations, as well as varios constraints of the onboard sensors and flight dinamics of the UAV. Both paths, being represented in compact mathematical form, are then transmitted onboard the networked UAV over the wireless link. The rest of the mission is performed by the flight controller onboard the UAV autonomously. The graphical illustration of the concept of autonomous feature following operation is illustrated in Fig. 1. This approach results in high quality reconnaissance data utilizing much smaller airspace footprint when compared to the conventional waypoint navigation solution. In addition, the latter can't guarantee that the sensor continuously maintains the feature path in the center of the camera frame with the specified resolution regardless of the shape of the geographical feature.

Majority of the commercial autopilots available for flight control of the UAVs employs traditional way-point (WP) navigation [4]. However, the WP navigation is not suited for aggressive and complex flight trajectory for visual target tracking applications since it might require long straight line segments for precise camera pointing [5]. Vision-based road following and flight navigation applications has been demonstrated using an onboard camera [6,7]. It requires an efficient real-time vision control system that detects natural features and tracks the feature road for the UAV with real-time image processing strategy, which is obtained at an additional onboard computational cost. These shortcomings can be overcome by implementing a new real-time path following control and camera control algorithms embedded onboard the UAV. Previous work [2] reported provides a comprehensive solution to the path following problem that includes real-time nonlinear path-following algorithms augmented by L_1 adaptive control. In this

Manuscript received February 28, 2010; accepted June 10, 2010. Recommended by Guest Editor Seong G. Kong. This work was supported by SOCOM under NPS-SOCOM TNT cooperative. The research of the first author is supported by the National Research Council Associateship tenured at the Center for Autonomous Vehicle Research at the Naval Postgraduate School.

Deok-Jin Lee is with the Center for Autonomous Vehicle Research (CAVR), Naval Postgraduate School, Monterey, CA, 93943, U.S.A. (e-mail: djlee@nps.edu, apollo17@gmail.com).

Isaac Kaminer, Vladimir Dobrokhodov, and Kevin Jones are with the Department of Mechanical and Astronautical Engineering, Naval Postgraduate School, CA 94943, U.S.A. (e-mails: {kaminer,vldobr, kdjones}@nps.edu).

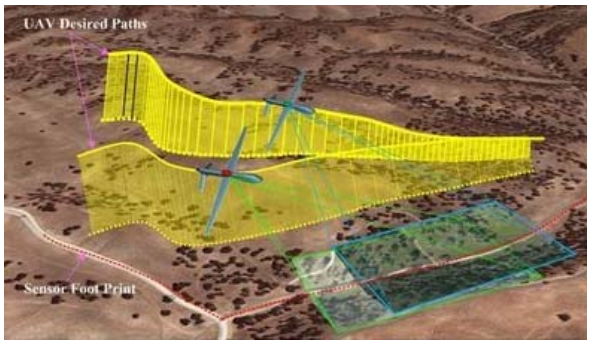

Fig. 1. Cooperative feature following with a camera onboard a small UAV as a flying image sensor.

paper, these algorithms are further extended to the feature following problem in a way that efficiently enables a user to select any geographical feature on a geo-referenced map to develop a surveilance mission for UAVs. The corresponding UAV flight path is obtained using a mapping technique which extracts points of the geographical feature and produces a dynamically feasible path in a geographical coordinate system.

Objectives of this paper are twofold. The first is to develop an efficient feature following control algorithm that guarantees presence of the selected geographical feature in the center of the camera image frame at all times. The second objective consists of developing an onboard camera control capability for the imaging sensor. The first objective is achieved by integrating an autonomous feature generation and optimization algorithm with the path following control technique. The resulting feasible flight path is independent of time that enables space and time separation and allows for near real-time onboard implementation. An optimization technique used to generate these flight paths maximizes the field of view of either a fixed or gimbaled camera onboard the UAV subject to its bank angle constraints. Once the optimized feature path is generated, it is sent to the onboard computer where it is used by the integrated path-following and navigation algorithms to control the UAV and the onboard camera. Sensor fusion technique for an efficient navigation algorithm with a cascaded filtering architecture is developed to provide a high-rate navigation solution for the feature following algorithm. The proposed navigation algorithm fuses the rate gyro information with the GPS measurements. The integration of the feature-following algorithm, reliable navigation, and camera stabilization along with the remote networking capability provide a simple and effective solution to the complex feature tracking problem of extracting high resolution imagery for efficient real-time visual surveillance.

The remainder of this paper is organized as follows. Section 2 describes the autonomous feature following algorithm. Section 3 discusses the development of a distributed navigation system that provides position velocity and attitude estimates. In Section 4 a summary of the new third-generation of rapid flight test prototyping system (RFTPS) developed in the Naval Postgraduate School is presented. Section 5 presents some of the flight test results.

2. AUTONOMOUS FEATURE FOLLOWING CONTROL

In this section, the generation of an optimized feature path is introduced. It is divided into three steps. In the first step, a feature path of a general road is extracted by using simple point-and-click or mouse dragging scribble operation on a geo-referenced digital map. In the second step, the extracted feature is enhanced by using a smoothing algorithm to clean out the jumps coming from jitter or mouse movement. Finally the smoothed feature path is optimized to obtain the analytical representation of the feasible feature defined by the 3D spatial polynomial that is a function of a virtual arc length. This overall process is illustrated in Fig. 2.

For instance, Fig. 3 shows a feature track extracted by using simple mouse scribble operation on a digital map. After the feature track on the map is enhanced by using a smoothing algorithm, it is approximated by a polynomial path shown in Fig. 4. The polynomial representation of the feature path is used to determine a flight path for the UAV. This step is cast as an optimization problem detailed in the remainder of this section.

Let $\mathbf{p}_c(\tau) = [x(\tau), y(\tau), z(\tau)]^T$ present the feature path where $\tau = [0, \tau_f]$ and τ_f is the total virtual arc length. Each coordiante $x(\tau)$, $y(\tau)$, $z(\tau)$ is represented by a *N*- degree of polynimoal [9]

$$x_i(\tau) = \sum_{d=0}^{N} a_{i,d}\,\tau^d, \quad i = 1,2,3, \tag{1}$$

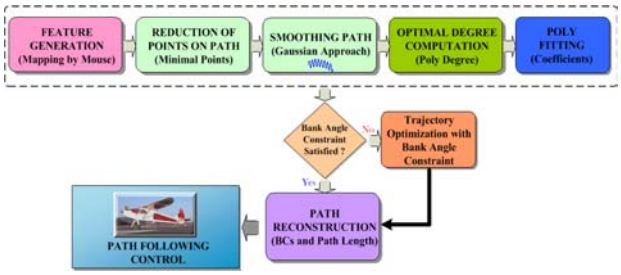

Fig. 2. Diagram for autonomous feature path generation.

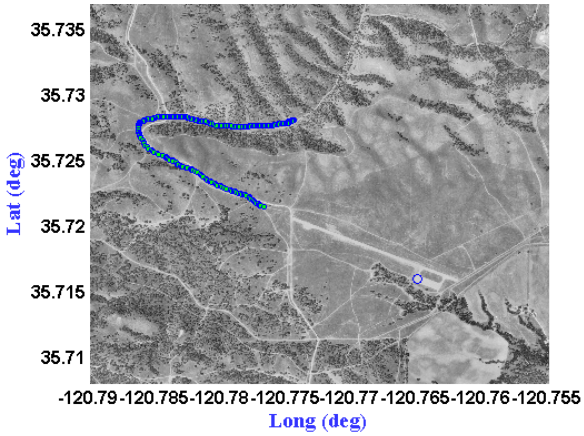

Fig. 3. Feature trajectory mapping by using point-and-click or dragging.

Fig. 4. Polynomial feature trajectory reconstruction in geographical coordinates.

where $(x_1 = x, x_2 = y, x_3 = z)$ for the sake of simplicity.

The degree of the polynomials is determined by the number of initial and final boundary conditions. Let N_0 and N_f represent the order of the highest derivative of $\mathbf{p}_c(0)$ and $\mathbf{p}_c(\tau_f)$, then the minimum degree of each polynomial is $N^* = N_0 + N_f + 1$. These in turn can be used to compute the polynomial coefficients in (1). For example, if $N_0 = N_f = 2$ then $N^* = 5$ and as a result (2) can be used to compute the coefficients of a 5th degree of polynomial path. (Details on this approach can be found in [9]).

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 1 & \tau_f & \tau_f^2 & \tau_f^2 & \tau_f^2 & \tau_f^2 \\ 0 & 1 & 2\tau_f & 3\tau_f^2 & 4\tau_f^2 & 5\tau_f^2 \\ 0 & 0 & 2 & 6\tau_f & 12\tau_f^2 & 20\tau_f^2 \end{bmatrix} \begin{bmatrix} a_{i,0} \\ a_{i,1} \\ a_{i,2} \\ a_{i,3} \\ a_{i,4} \\ a_{i,5} \end{bmatrix} = \begin{bmatrix} x_i(0) \\ x_i'(0) \\ x_i''(0) \\ x_i(\tau_f) \\ x_i'(\tau_f) \\ x_i''(\tau_f) \end{bmatrix} \tag{2}$$

The parameterization in (1) completely determines the 3D spatial UAV path with all the boundary conditions satisfied. Furthermore, from (2) it is clear that the total virtual arc length τ_f can be used as an optimization parameter.

Given the position vector in (1), the curvature of the feature path is calculated by [1]

$$\kappa \equiv \frac{1}{r} = \left\| \frac{d\mathbf{T}}{d\tau} \right\| \frac{1}{\left\| \mathbf{p}'(\tau) \right\|}, \tag{3}$$

where r is a local radius and

$$\mathbf{T} = \mathbf{p}'(\tau) \frac{1}{\left\| \mathbf{p}'(\tau) \right\|}, \quad \mathbf{p}'(\tau) = \frac{d\mathbf{p}(\tau)}{d\tau}. \tag{4}$$

In a typical feature following mission the UAV is flying along the feature path with a constant velocity $v = \|\mathbf{v}\|$. In this case the bank angle of the UAV along the path ϕ_{UAV} is given by

$$\phi_{UAV} = \tan^{-1}\left(\frac{v^2}{gr} \right) = \tan^{-1}\left(\frac{v}{g} \kappa \right). \tag{5}$$

Typical constraints on the bank angle and roll rate of the UAV are

$$0 < \left| \phi_{UAV} \right| \le 35^\circ, \quad \left| \dot{\phi}_{UAV} \right| \le \dot{\phi}_{\max}, \tag{6}$$

where the roll rate angle is computed by

$$\dot{\phi} = \frac{d\phi}{d\tau} \frac{d\tau}{dt} = \frac{d\phi}{d\tau} \frac{v}{\left\| \mathbf{p}'(\tau) \right\|}. \tag{7}$$

For the case of constant velocity the acceleration along the flight path must satisfy the following constraint

$$\mathrm{a}_p(\tau) \le a_{\max}, \tag{8}$$

where

$$\mathrm{a}_p(\tau) = \frac{v_p^2}{\left\| \mathbf{p}'(\tau) \right\|^2} \left\| \left(I - \frac{\mathbf{p}'(\tau)\left(\mathbf{p}'(\tau)\right)^T}{\left\| \mathbf{p}'(\tau) \right\|^2} \right) \mathbf{p}''(\tau) \right\|. \tag{9}$$

Next using the problem geometry shown in Fig. 5 we obtain additional constraints on the UAV bank angle that guarantee that the user specified feature remains in the center of the image frame of the onboard camera. Let ϕ_T denote the angle between the line connecting the origin of the UAV fixed frame to the feature and local horizon. Let $\mathbf{p}_T(\tau)$ denote the feature path. Then

$$\phi_T = \tan^{-1}\left(\frac{i_y}{i_z} \right), \tag{10}$$

where

$$\frac{\mathbf{p}_{UAV}(\tau) - \mathbf{p}_T(\tau)}{\left\| \mathbf{p}_{UAV}(\tau) - \mathbf{p}_T(\tau) \right\|} = \begin{bmatrix} i_x \\ i_y \\ i_z \end{bmatrix}. \tag{11}$$

From Fig. 5 it follows that the desired feature will remain at the center of the image frame if

$$\phi_T = \frac{\pi}{2} - \left(\phi_{UAV} + \phi_G \right), \tag{12}$$

where ϕ_G is the angle of the gimbaled camera between the UAV vertical frame to the line of sight to the feature. On the other hand, if

$$\left| \phi_T - \frac{\pi}{2} + \left(\phi_{UAV} + \phi_G \right) \right| \le \varepsilon < \vartheta / 2, \tag{13}$$

where ϑ is the LOS of the camera and ε is a threshold value to be minimized, then the feature will remain in the image frame of the onboard camera. Therefore, determining a desired flight path for the UAV can be reduced to the following optimization problem

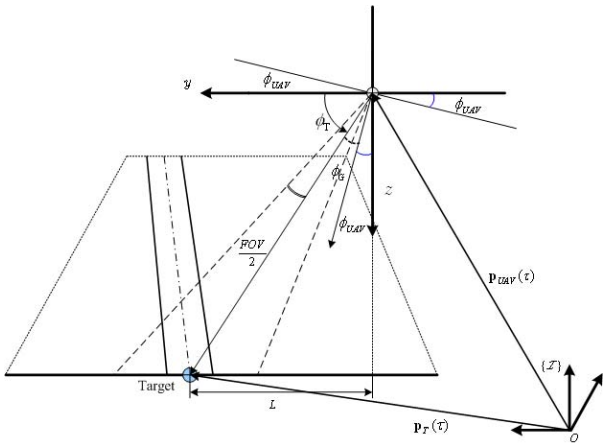

Fig. 5. Geometry between bank angles and lateral distance errors.

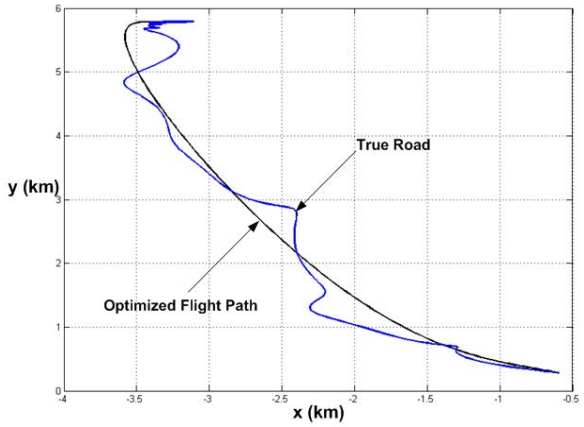

Fig. 6. Modified new flight trajectory generated from the optimization algorithm.

$$\arg\min_{v,\tau_f} \varepsilon \tag{14}$$

subject to the following constraints

$$0 < |\phi_{UAV}| \le \phi_{\max}, \quad |\dot{\phi}_{UAV}| \le \dot{\phi}_{\max}, \tag{15}$$

$$\left|\phi_T - \frac{\pi}{2} + (\phi_{UAV} + \phi_G)\right| \le \varepsilon < \vartheta/2, \tag{16}$$

$$v_{\min} \le v_p(\tau) \le v_{\max}, \; a_p(\tau) \le a_{\max}. \tag{17}$$

Fig. 6 illustrates a solution to the optimization problem in (14). It includes a reference feature path and an optimized flight path for the UAV (modified feature path) obtained by solving (14). After an optimized flight path is generated, path following control algorithms onboard UAVs are executed to make each vehicle to follow the generated spatial path (See [2] for detail).

3. DISTRIBUTED NAVIGATION SYSTEMS DESIGN

The primary objective of this section is to design a multi-rate navigation filter to provide precise estimates of the UAV position and velocity with respect to a local geographical navigation frame (North, East, Down). This is done using fault-tolerant distributed cascaded filtering architecture while keeping the computational workload as low as possible.

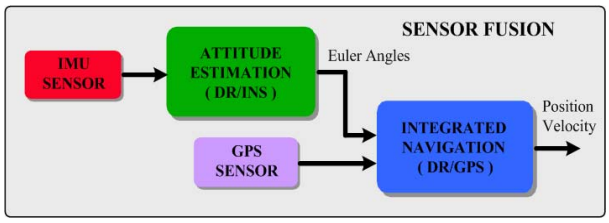

Fig. 7. Diagram for distributed navigation system with cascade filtering structure.

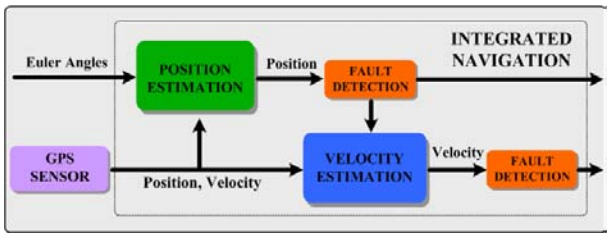

Fig. 8. Diagram for integrated position and velocity estimation.

Fig. 7 shows the overall distributed navigation concept for attitude, position and velocity estimation using cascaded filtering architecture. First, a complementary filter [10] is used to estimate attitude of the UAV. Second, an integrated dead-reckoning and GPS navigation system that combines a low-cost strapdown inertial rate data and GPS measurement by using the extended Kalman filter is designed. A simple dead-reckoning (DR) approach is used to provide prediction of the position and velocity in between GPS updates. The proposed integrated DR/GPS algorithm minimizes the onboard computational workload.

The architecture of the position and velocity estimation filter is shown in Fig. 8 where the navigation algorithms are designed by integrating the attitude information from the attitude complementary filter with the position and velocity from the GPS sensor. A detailed derivation of this architecture is presented in below.

3.1. Position estimation

Suppose the velocity components of the vehicle resolved in the body frame are given, then the equation of motion of the UAV in the navigation frame is computed by multiplying a direction cosine matrix provided from the attitude estimation subsystem with the body velocity vector as follows [11]

$$\dot{\mathbf{p}}^n = \mathbf{C}_b^n \mathbf{v}^b, \tag{18}$$

where $\mathbf{p}^n \in \Re^{3\times1}$ represents the position vector expressed in the local geographic navigation frame defined by the directions of north, east , and down, $\mathbf{p}^n = [p_N \quad p_E \quad -h]^T \in \Re^{3\times1}$ and $\mathbf{v}^b \in \Re^{3\times1}$ denotes the velocity vector in the body frame with directions, out the noise, the right wing, and the down belly, respectively,

$\mathbf{v}^b \equiv [u\ v\ w]^T$. Since the attitude and navigation filtering structure is based on the cascade form, the direction cosine matrix, $\mathbf{C}_b^n \in \Re^{3\times3}$, is available from the attitude estimation in advance. Now the continuous form of the position navigation equation can be discretized as

$$\mathbf{p}_{k+1}^n = \mathbf{p}_k^n + \int_{t_k}^{t_{k+1}} \mathbf{C}_b^n \mathbf{v}^b dt \approx \mathbf{p}_k^n + \Delta t \mathbf{C}_b^n \mathbf{v}^b. \tag{19}$$

Suppose the direction cosine matrix is parameterized in terms of quaternion, $\mathbf{q} \equiv [\boldsymbol{\varrho}^T \quad q_4]$, then the discrete position equation can be written in a form of a nonlinear function as $\mathbf{p}_{k+1}^n = \mathbf{f}_p(\mathbf{p}_k^n, \mathbf{q}_k, \mathbf{v}^b)$ where the nonlinear function $\mathbf{f}_p$ is a function of a position vector, a quaternion vector, and a body frame velocity vector. Usually, the computation of the position can be made in a computationally efficient way that the body velocity component can be replaced with a speed based on the assumption that there is negligible sideslip and angle of attack. Therefore, the forward velocity vector, u, can be replaced with the speed of the UAV, V, and the others are zeros, i.e., $u \approx V$, $v = w \approx 0$. The body velocity vector can be written as $\mathbf{v}^b \equiv [V\ 0\ 0]^T$.

If there exists an estimate $\hat{\mathbf{p}}_k^n \in \Re^{3\times1}$ at time k, then the predicted position estimate at time $k+1$ is approximated by

$$\hat{\mathbf{p}}_{k+1|k}^n = \mathbf{f}_p(\hat{\mathbf{p}}_k^n, \hat{\mathbf{q}}_{k|k}, \mathbf{v}^b) = \hat{\mathbf{p}}_{k|k}^n + \Delta t \mathbf{C}_b^n(\hat{\mathbf{q}}_{k|k})\mathbf{v}^b, \tag{20}$$

where the quaternion estimate, $\hat{\mathbf{q}}_{k|k}$, is provided by the attitude complementary filter. Now the predicted state covariance equation is computed by the outer product of the predicted estimate error

$$\mathbf{P}_{k+1|k}^p = E[\boldsymbol{\delta}\mathbf{p}_{k+1|k} \boldsymbol{\delta}\mathbf{p}_{k+1|k}^T \,|\, (\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_k)], \tag{21}$$

where $\boldsymbol{\delta}\mathbf{p}_{k+1|k}$ is the predicted state estimate error defined by subtracting the true state given in (19) from the predicted state given in (20) as follows $\boldsymbol{\delta}\mathbf{p}_{k+1|k} \equiv \mathbf{p}_{k+1}^n - \hat{\mathbf{p}}_{k+1|k}^n$. After expanding (20) using the first-order Taylor series about the estimate, $\hat{\mathbf{p}}_{k|k}^n$, the predicted position estimate error is approximated by

$$\boldsymbol{\delta}\mathbf{p}_{k+1|k} \approx \left.\frac{\partial \mathbf{f}_p}{\partial \mathbf{p}_{k|k}^n}\right|_{\left(\mathbf{p}_k^n - \hat{\mathbf{p}}_{k|k}^n\right)} \boldsymbol{\delta}\mathbf{p}_{k|k}, \tag{22}$$

where $\mathbf{F}_{p,k} \in \Re^{3\times3}$ is the Jacobian matrix of $\mathbf{f}_p$ evaluated at $\mathbf{p}_{k|k}^n = \hat{\mathbf{p}}_{k|k}^n$. It should be noted that the position prediction given in (20) has a nonlinear term, $\mathbf{f}_{p,2}(\hat{\mathbf{q}}_k, \mathbf{v}^b) \equiv \Delta t \mathbf{C}_b^n(\mathbf{q}_k)\mathbf{v}^b$, which is a function of the estimated quaternion and the body velocity vector rather than the position estimate vector. This makes it difficult to derive the Jacobian matrix, $\mathbf{F}_{p,k}$, directly. However, the Jacobian matrix can be derived by applying the chain rule and having the assumption that the estimated quaternion and velocity information is fixed over the computational cycle $\Delta t = t_{k+1} - t_k$. It is assumed that the augmented state vector $\hat{\boldsymbol{u}}_k$ is defined, $\hat{\boldsymbol{u}}_k \equiv [\hat{\mathbf{q}}_k^n\ \mathbf{v}^b]$, then, the Jacobian matrix can be approximated as follows

$$\begin{aligned}\mathbf{F}_{p,k} &= \left.\frac{\partial \mathbf{f}_p}{\partial \mathbf{p}_{k|k}^n}\right| = \mathbf{I}_{3\times3} + \left.\frac{\partial \mathbf{f}_{p,2}}{\partial \hat{\boldsymbol{u}}_k}\frac{\partial \hat{\boldsymbol{u}}_k}{\partial \mathbf{p}_{k|k}^n}\right| \\ &= \mathbf{I}_{3\times3} + \mathbf{F}_{2,k}\left.\frac{\partial \hat{\boldsymbol{u}}_k}{\partial \mathbf{p}_{k|k}^n}\right|.\end{aligned} \tag{23}$$

Then, the predicted position covariance equation is approximated by

$$\mathbf{P}_{k+1|k}^p \approx \mathbf{P}_{k|k}^p + \mathbf{F}_{2,k}\boldsymbol{\mathcal{P}}_k^u \mathbf{F}_{2,k}^T, \tag{24}$$

where $\boldsymbol{\mathcal{P}}_k^u = E[\boldsymbol{\delta}\hat{\boldsymbol{u}}_k \boldsymbol{\delta}\hat{\boldsymbol{u}}_k^T] \in \Re^{7\times7}$ can be divided into two parts. One is the covariance matrix based on the quaternion and the other is derived in terms of the velocity vector. Since the quaternion part of $\hat{\boldsymbol{u}}_k$ is the output obtained from the attitude filter that produces the steady-state attitude estimate with bounded estimation errors, the variance of $\hat{\boldsymbol{u}}_k$ can be assumed to be a steady-state value, i.e., constant covariance matrix, $\boldsymbol{\mathcal{P}}_k^p = \boldsymbol{\mathcal{P}}^p$, is composed of two block diagonal matrices

$$\boldsymbol{\mathcal{P}}_k^p \equiv \begin{bmatrix} \boldsymbol{\mathcal{P}}_q^p & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\mathcal{P}}_v^p \end{bmatrix}. \tag{25}$$

This will be a tuning variable that is specified based on the attitude sensor systems to make the position estimation accurate.

The measurement equation is expressed in a linear form by

$$\tilde{\mathbf{y}}_{p,k+i} = \mathbf{p}_{k+m}^n, \quad \begin{cases} i = 1, \ldots, \infty \\ m = \mathrm{int}\left(i/ds\right), \end{cases} \tag{26}$$

where *int* operator produces an integer value, *ds* is the sampling rate, i.e., $ds = 10Hz$, the measurement matrix $\mathbf{H}_{p,k}$ becomes diagonal, and it is assumed that the GPS sensor noise vector with covariance matrix, $\mathbf{v}_{p,k} \sim (\mathbf{0}, \mathbf{R}_{p,k})$. Suppose there exists a predicted position estimate $\hat{\mathbf{p}}_{k+1|k}^n \in \Re^{3\times1}$ at time $k+1$, then the innovation vector, $\boldsymbol{\upsilon}_{p,k+1}$, which is the difference between the sensor measurement and the predicted observation at time $k+1$, is defined as $\boldsymbol{\upsilon}_{p,k+1} \equiv \tilde{\mathbf{y}}_{p,k+1} - \hat{\mathbf{y}}_{p,k+1}$. The covariance of the innovation vector is obtained by [12]

$$\mathbf{P}_{p,k+1}^{\upsilon\upsilon} = \mathbf{H}_{p,k+1}\mathbf{P}_{k+1|k}^p \mathbf{H}_{p,k+1}^T + \mathbf{R}_{p,k}. \tag{27}$$

Now, the Kalman gain matrix is computed by

$$\mathcal{K}_{k+1}^{p} = \mathbf{P}_{k+1|k}^{p}\mathbf{H}_{p,k+1}^{T}\left(\mathbf{P}_{p,k+1}^{vv}\right)^{-1}, \tag{28}$$

where $\mathbf{P}_{k+1}^{py} \equiv \mathbf{P}_{k+1|k}^{p}\mathbf{H}_{p,k+1}^{T}$ is the predicted cross-correlation matrix between the predicted position state and measurement vector. Finally, the position estimate and covariance updates are obtained by

$$\hat{\mathbf{P}}_{k+1|k+1}^{n} = \hat{\mathbf{P}}_{k+1|k}^{n} + \mathcal{K}_{k+1}^{p}\boldsymbol{\upsilon}_{p,k+1}, \tag{29}$$

$$\mathbf{P}_{k+1|k+1}^{p} = \mathbf{P}_{k+1|k}^{p} - \mathcal{K}_{k+1}^{p}\mathbf{P}_{k+1}^{vv}(\mathcal{K}_{k+1}^{p})^{T}. \tag{30}$$

3.2. Velocity estimation

In general, the contribution of the Coriolis acceleration terms, and also the effect of the earth angular velocity, the centripetal force, is small for navigation in the vicinity of the Earth. After simplification, the navigation equations of a UAV in the local geographic navigation frame is expressed by [13]

$$\dot{\mathbf{v}}^{n} = \mathbf{f}^{n} + \mathbf{g}^{n}, \tag{31}$$

where $\mathbf{f}^{n}$ is the specific force vector measured by a triad of accelerometers resolved in the navigation frame, $\mathbf{f}^{n} = \mathbf{C}_{b}^{n}\mathbf{f}^{b}$, and $\mathbf{g}^{n}$ is the gravitational acceleration contribution in the navigation frame, $\mathbf{g}^{n} = [0\ 0\ \mathrm{g}]^{T}$. Then, the continuous navigation equation is converted into the discrete-time form by

$$\mathbf{v}_{k+1}^{n} = \mathbf{v}_{k}^{n} + \mathbf{u}_{f,k}^{n} + \mathbf{g}^{n}\Delta t, \tag{32}$$

where $\mathbf{u}_{f,k}^{n}$ represents the sum of the velocity changes over each update, and is a function of vehicle body attitude, $\mathbf{f}^{n} = \mathbf{C}_{b}^{n}\mathbf{f}^{b}$.

$$\mathbf{u}_{f,k}^{n} = \int_{t_k}^{t_{k+1}}\mathbf{f}^{n}dt = \int_{t_k}^{t_{k+1}}\mathbf{C}_{b}^{n}\mathbf{f}^{b}dt \tag{33}$$

Now, the discrete time velocity equation is written in a nonlinear function as $\mathbf{v}_{k+1}^{n} = \mathbf{f}_{v}(\mathbf{v}_{k}^{n}, k)$. Now, the predicted velocity estimate and state covariance at time $k+1$ is approximated by

$$\hat{\mathbf{v}}_{k+1|k}^{n} = \hat{\mathbf{v}}_{k|k}^{n} + \mathbf{u}_{f,k}^{n} + \mathbf{g}^{n}\Delta t, \tag{34}$$

$$\mathbf{P}_{k+1|k}^{v} \approx \mathbf{P}_{k|k}^{v} + \mathbf{F}_{2,k}^{v}\mathcal{P}_{k}^{v}\mathbf{F}_{2,k}^{T}, \tag{35}$$

where the Jacobian matrix $\mathbf{F}_{2,k}^{v}$ is given as follows

$$\mathbf{F}_{2,k}^{v} = \left.\frac{\partial\mathbf{f}_{v,2}}{\partial\hat{\boldsymbol{u}}_{v,k}}\right|. \tag{36}$$

$\mathbf{f}_{v,2}(\hat{\mathbf{q}}_{k}, \mathbf{f}^{b}) \equiv \mathbf{u}_{f,k}^{n} = \Delta t\mathbf{C}_{b}^{n}(\mathbf{q}_{k})\mathbf{f}^{b}$ is the velocity change term, and $\hat{\boldsymbol{u}}_{v,k}$ is the augmented state vector defined, $\hat{\boldsymbol{u}}_{v,k} \equiv [\hat{\mathbf{q}}_{k}^{n}\ \mathbf{f}^{b}]$. Then, $\mathbf{F}_{2,k}^{v}$ becomes equal to the jacobian matrix of the position part, i.e., $\mathbf{F}_{2,k}^{v} = \mathbf{F}_{2,k}^{p}$.

$\mathcal{P}_{k}^{v} = E[\delta\hat{\boldsymbol{u}}_{v,k}\delta\hat{\boldsymbol{u}}_{v,k}^{T}] \in \Re^{3\times 7}$ is a scaled weighted matrix that is composed of two block diagonal matrices

$$\mathcal{P}_{k}^{v} \equiv \begin{bmatrix}\mathcal{P}_{q}^{v} & \mathbf{0}\\ \mathbf{0} & \mathcal{P}_{a}^{v}\end{bmatrix}. \tag{37}$$

Since the first part is connected to the quaternion output from the attitude filter that produces the steady-state attitude estimate, the diagonal matrix has the same covariance value in the position estimation, $\mathcal{P}_{q}^{v} = \mathcal{P}_{q}^{p}$. The second block diagonal, $\mathcal{P}_{a}^{v}$, is a tunable weight matrix. Since the velocity vector, $\mathbf{v}_{k}^{n}$, is the product of the direction cosine $\mathbf{C}_{b}^{n}$ and the body velocity (or the speed V) of the UAV as, $\mathbf{v}_{k}^{n} = \mathbf{C}_{b}^{n}\mathbf{v}^{b}$, the predicted covariance, $\mathbf{P}_{k|k}^{v}$, of the velocity term can be replaced by the second term in the position covariance. Thus, the predicted velocity error covariance equation can be simplified further as

$$\mathbf{P}_{k+1|k}^{v} \approx \frac{1}{\Delta t}\mathbf{F}_{2,k}\left[\mathcal{P}_{k}^{p} + \Delta t\mathcal{P}_{k}^{v}\right]\mathbf{F}_{2,k}^{T}, \tag{38}$$

where $\mathbf{F}_{2,k} \equiv \mathbf{F}_{2,k}^{p} = \mathbf{F}_{2,k}^{v}$. In a similar way used in the position estimation, the velocity measurement equation can be expressed

$$\tilde{\mathbf{y}}_{p,k+i} = \mathbf{v}_{k+m}^{n}, \quad \begin{cases} i = 1,\ldots,\infty \\ m = \mathrm{int}\left(i/ds\right). \end{cases} \tag{39}$$

Then, the velocity estimate and covariance updates are obtained by applying the Kalman update equation given in (29)-(30) as did in the position estimation.

3.3. Sensor fault detection

The distributed navigation system implemented is based on the cascaded senor information integration which provides sensor failure detection and isolation capability by isolating any anomalous sensor data resulting from sensor failure or from corruption of the sensor signals [8]. For sensor fault detection, the statistical chi-test χ^{2} is utilized. First, the associated likelihood function for innovation is defined by

$$\mathcal{L}\left\{\boldsymbol{\upsilon}_{k+1}\right\} = \exp\left\{-\frac{1}{2}\boldsymbol{\upsilon}_{k+1}^{T}\left(\mathbf{P}_{k+1}^{vv}\right)^{-1}\boldsymbol{\upsilon}_{k+1}\right\}, \tag{40}$$

where $\boldsymbol{\upsilon}_{k+1}$ is the measurement innovations vector and the innovation covariance matrix is given by the navigation filter as

$$\mathbf{P}_{k+1}^{vv} = \mathbf{H}_{k+1}\mathbf{P}_{k+1|k}\mathbf{H}_{k+1}^{T} + \mathbf{R}_{k}. \tag{41}$$

Now, the log-likelihood is given by

$$\log\mathcal{L}\left\{\boldsymbol{\upsilon}_{k+1}\right\} = -\boldsymbol{\upsilon}_{k+1}^{T}\left(\mathbf{P}_{k+1}^{vv}\right)^{-1}\boldsymbol{\upsilon}_{k+1}. \tag{42}$$

Then the normalized innovation is expressed by

$$d_{k+1}^2 \equiv \boldsymbol{v}_{k+1}^T \left(\mathbf{P}_{k+1}^{vv}\right)^{-1} \boldsymbol{v}_{k+1}, \tag{43}$$

which is a normalized innovation describes a quadratic ellipsoidal volume centered on the observation prediction. If an observation falls within this volume, then it is considered valid. For this test the equivalent statistical χ^2 test is expressed by

$$\chi^2 = \frac{\boldsymbol{v}_{k+1}^T \left(\mathbf{P}_{k+1}^{vv}\right)^{-1} \boldsymbol{v}_{k+1}}{n_v} \geq 0, \tag{44}$$

where n_v is the dimension of the innovation vector. This has non-negative value with a minimum value of zero, thus an upper limit threshold value on χ^2 can be used to detect anomalous sensor information. A threshold value $\chi^2_{\max}$ is chosen such that the sensor data is rejected when $\chi^2 > \chi^2_{\max}$.

4. UAV FLIGHT SYSTEMS

The schematic diagram of overall concept of the feature following control system is shown in Fig. 9 where the control inputs from the feature following algorithm are fed back to the autopilot as an outer loop control and for an inner loop controller, an L_1 adaptive control law can be applied to compensate for a worst situation or for enhancing the feature following performance [2].

For feature tracking application a Rascal UAV as a rapid flight test prototyping system (RTFPS) for small unmanned air vehicles (SUAVs) developed at the Center for Autonomous Vehicle Research in the Naval Postgraduate School is utilized for hardware-in-the-loop simulation and real flight test experiment [4]. The new RTFPS integrated avionics system architecture that includes all the principal components along with the Piccolo plus

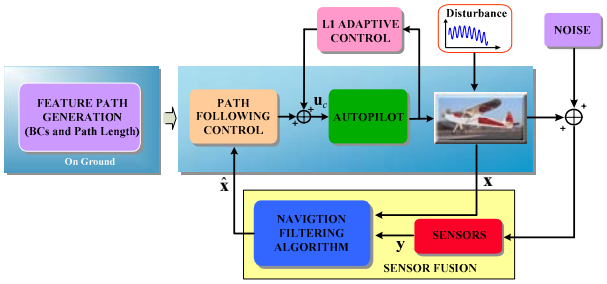

Fig. 9. Diagram for integrated position and velocity estimation.

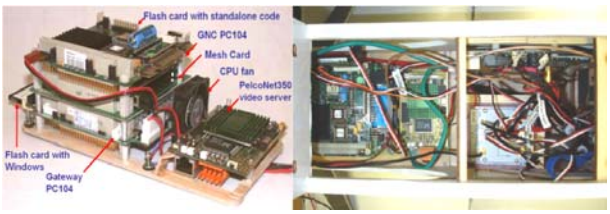

Fig. 10. Hardware integration and implementation architecture.

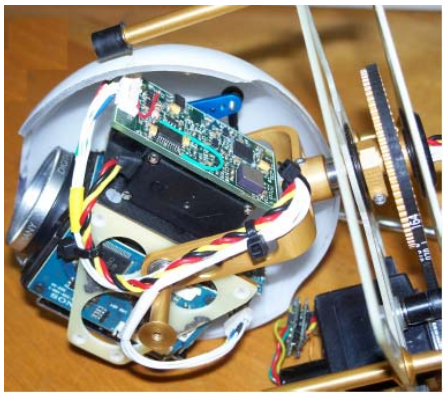

Fig. 11. Gimbaled camera unit with 10x zoom.

autopilot [14], and an overhead view of the avionics is shown in Fig. 10.

The Piccolo plus autopilot is used for primary flight control, with its dedicated 900MHz serial data link. In the integrated flight system, two *PC104* [15] computers are integrated, one is for a task of guidance, navigation and control (GNC), and the other plays a role of a gateway computer to bridge an onboard LAN and a wireless mesh networking. A Persistent Systems Wave Relay router [16] is optimized for mobile ad-hoc broadband networking. A Pelco NET300T video server [17] is used to stream the analog video feed from the camera, and all network devices are linked through a Linksys 5-port hub. Analog image took from a CCD camera is transmitted through the PelcoNet video server (NET300T) across an Ethernet networks integrated in the SUAV system to a ground control center. The NET300T can display the video on a PC through any Web browser. The integrated avionics package weighs total 1 lb, and requires about 25 W power with all components active.

Imagery is provided by a Sony FCB-IX11A color block-camera [18] shown in Fig. 11 which has the features; a 1/4 CCD, 10x optical zoom, 1-9/16 x 1-13/16 x 2-5/8 inches measurement, 1.5 W low power consumption, and 3.5 ounces weight, and is ideal for space-limited small UAVs. It provides the on-line features such as zoom, auto focus, adjustable gain, white balancing, and titling capabilities available over the high-speed serial communication link and TTL signal-level control for quick command processing.

The camera is mounted in a low-cost 2-axis gimbaled system that has a 10-cm ball with the lower half exposed under the belly of the UAV just behind wings. The gimbaled system has the pan and tilt channels driven by two high-speed digital servos which are actuated through a serial-PWM 12 bit controller connected to the RISC ATMEGA-169 8-bit microcontroller (8MHz) which is the central component of the architecture that implements gimbal control through integration of line-of-sight (LOS) rate measurements (tri-axial sensor head), and gimbal reference commands sent from onboard GNC SBC PC104 computer [4]. The LOS inertial stabilization technique is based on the subtraction of the UAV Euler angular rates measured in inertial space from the gimbal reference commands. The schematic diagram of the gim-

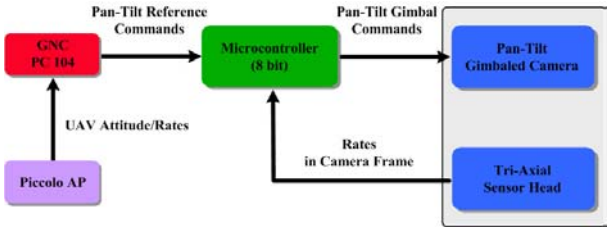

Fig. 12. Diagram of inertial stabilization scheme in gimbaled camera unit.

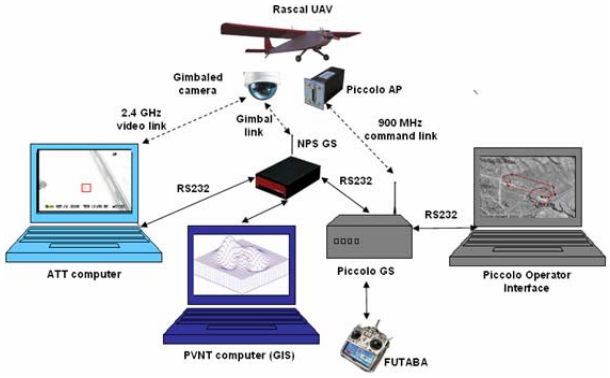

Fig. 13. Flight control and feature following control system for visual surveillance.

baled system is illustrated in Fig. 12.

Fig. 13 depicts the architecture of the feature path tracking strategy. When the UAV receives a feature following command from a ground control center, the UAV and the camera automatically start following the polynomial feature path for surveillance and tracking of a target that is moving along the feature trajectory by using the information provided from the Piccolo ground control center and onboard path following control algorithm. The integrated UAV/gimbaled control algorithm in turn keeps the tracking in the center of the camera frame. The video imagery taken from the gimbaled camera is transmitted to the image processing computer over the analog 2.4 GHz link in real-time. For the inner loop controller, an L_1 adaptive control law can be applied to compensate for a worst situation or for enhancing the feature following performance.

5. FLIGHT TEST RESUTLS

This flight test was conducted in support of the quarterly, joint Cooperative U.S. Special Operations Command and the Naval Postgraduate School Field Experiments, held at McMillan field in Camp Roberts California. In this section, sample results from several experiments are shown providing some insight into the capabilities of the feature following system.

Fig. 14 shows a desired feature path generated by the mapping algorithm, which is reconstructed by using analytical polynomial equations as shown in Fig. 15. This polynomial path and its boundary conditions are passed onto the onboard computer to make the UAV track the desired path while keep the onboard camera targeting the feature. The overall feature following scenario is illu-

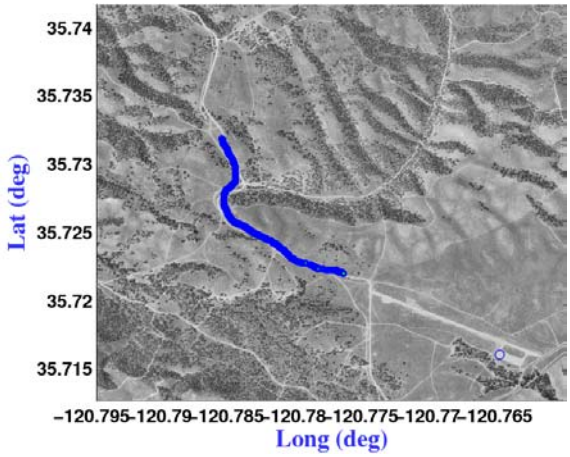

Fig. 14. Mapping for feature trajectory generation in geographical coordinates.

Fig. 15. Polynomial feature trajectory reconstruction in geographical coordinates.

strated in Fig. 16. Initially the small UAV is loitering near the ground control center with a constant radius at a predefined location. If a feature following command is sent to the UAV from the ground control station along with the information regarding the target feature path, the onboard computer generate an initial bridge trajectory, which provides a route for the UAV to reach the target feature from the current location. When the UAV reaches the target feature path, automatic transition command is executed to switch the UAV from the bridge trajectory to the target feature path. After the feature following is finished, the UAV is sent to the original loitering circle, where it repeats the tracking mission again.

The feature following errors between the commands and true UAV track is shown in Fig. 17, and the mean value of the deviation errors reside within 40 m and the maximum deviation from the desired trajectory is about 50 m during the feature following application. Real-time video image was obtained from the surveillance operation with the gimbaled camera system onboard the UAV. In addition to the visual information, static image can be constructed, and Fig. 18 shows the mosaic of the user selected features constructed from the images obtained by the onboard high resolution camera, taken at 550m AGL with a 30 degree FOV.

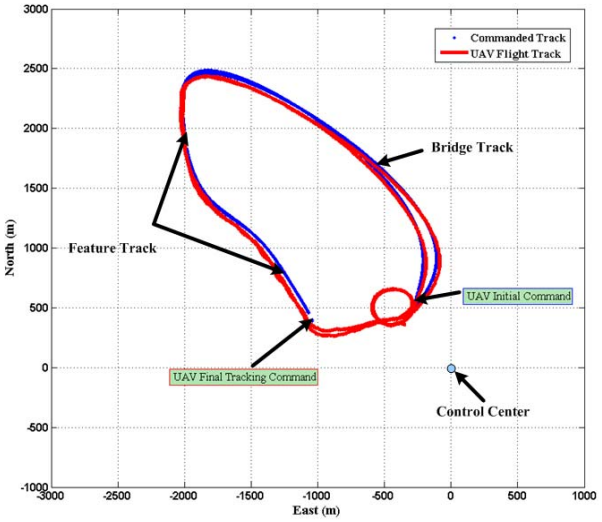

Fig. 16. Autonomous feature following results; blue line-commanded reference path, red line-UAV flight path.

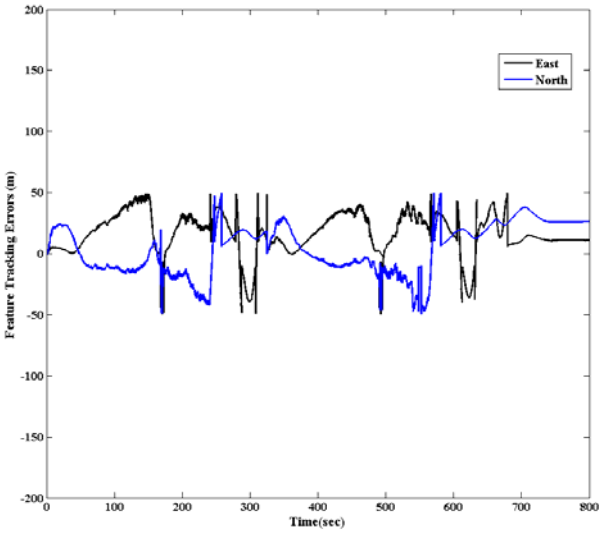

Fig. 17. Autonomous feature following errors in North and East coordinates.

Fig. 18. Mosaic obtained from images taken during the feature following application.

6. CONCLUSION

In this paper, a small UAV equipped with a low-cost, high-resolution imaging camera is utilized for feature following application. Flight test experimental results were presented. The newly integrated feature generation and following concept allows an operator with no prior training in UAV operations to specify the desired ground track to be followed by the airborne high-resolution camera, while the actual UAV flight path required to follow this sensor path is autonomously computed by the onboard computer. In addition, the proposed distributed navigation system provides very precise position and velocity estimates. Thus the complete system combines feature generation and following algorithms with advanced distributed navigation algorithm. This system was successfully used to capture imagery of discrete targets as well as continuous paths, where the feature following algorithm guaranteed that the sensor was pointing at the desired path. The developed feature-following algorithm, feature path generation and control algorithm, integrated dual-purpose HR sensor (still frames and full motion video) and remote networking capabilities provide a simple and effective solution to the complex autonomous path following problem enabling a high resolution video/imagery in surveillance and reconnaissance applications.

REFERENCES

[1] D. A. Schoenwald, "AUVs: in space, air, water, and on the ground," *IEEE Control Systems Magazine*, vol. 20, no. 6, pp. 15-18, 2000.

[2] I. Kaminer, O. Yakimenko, V. Dobrokhodov, A. Pascoal, N. Hovakimyan, A. Young, C. Cao, and V. Patel, "Coordinated path following for time-critical missions of multiple UAVs via L1 adaptive output feedback controllers," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Hilton Head, SC, 2007.

[3] M. E. Campbell, J.-W. Lee, and E. Scholte, "Simulation and flight test of autonomous aircraft estimation, planning, and control algorithms," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 6, pp. 1597-1609, 2007.

[4] K. Jones, V. Dobrokhodov, I. Kaminer, D.-J. Lee, E. Bourakov, and M. Clement, "Development, system integration and flight testing of a high-resolution imaging system for small unmanned aerial systems," *Proc. of the 47th AIAA Aerospace Sciences Meeting*, Orlando, Florida, Jan. 5-8 2009.

[5] R. Rysdyk, "Unmanned aerial vehicle path following for target observation in wind," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 29, no. 5, pp. 1092-1100, 2007.

[6] E. Frew, T. Mc-Gee, Z. Kim, X. Xiao, S. Jackson, M. Morimoto, S. Rathinam, J. Padial, and R. Sengupta, "Vision-based road following using a small autonomous aircraft," *Proc. of the IEEE Aerospace Conference*, pp. 3006-3015, 2004.

[7] J. Egbert and R. W. Beard, "Low altitude road fol-

lowing control constraints using strap-down EO cameras on miniature air vehicles," *Proc. of the IEEE American Control Conference*, pp. 353-358, 2007.
[8] M. S. Grewal, A. P. Andrews, and L. R. Weill, *Global Positioning Systems, Inertial Navigation*, John Wiley & Sons, Inc., New York, 2007.
[9] O, Yakimenko, "Direct method for rapid prototyping of near-optimal aircraft trajectories," *AIAA Journal of Guidance*, *Control, and Dynamics*, vol. 23, no. 5, pp. 865-875, 2000.
[10] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering,* 3rd ed., John Wiley & Sons, Inc., New York, NY, 1997.
[11] D. B. Kingston and R. W. Beard, "Real-time attitude and position estimation for small UAVs using low-cost sensors," *Proc. of AIAA Unlimited Systems Conference and Workshop*, Chicago, IL, Paper. No. AIAA-2004-6533, 2004.
[12] R. van der Merwe, E. A. Wan, and S. J. Julier, "Sigma-point Kalman filters for nonlinear estimation and sensor fusion: applications to integrated navigation," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, Rhode Island, 2004.
[13] D. H. Titterton and J. L. Weston, "Strapdown inertial navigation technology," *IEE Radar, Sonar, Navigation and Avionics Series 5*, London, UK, 1997.
[14] Piccolo Documentation, http://www.cloudcaptech.com.
[15] Microspace PC-104, http://www.adlogic-pc104.com.
[16] Wave Relay QUAD Radio Router, http://www.persistentsystems.com/products/.
[17] PelcoNet Video Server, http://www.pelco.com/producets.
[18] Sony FCB-IX11A Color Bock-Camera, http://www.aegis-elec.com/producets.

Deok-Jin Lee received his Ph.D. degree in Aerospace Engineering from Texas A&M University in May 2005. He worked for Agency for Defense Development (ADD) from 2006 to 2007. He is currently an adjunct research professor at the Center for Autonomous Vehicle Research, Naval Postgraduate School, Monterey, CA, U.S.A. His research interests include autonomous robotic vehicle control, sensor fusion and sensor networks, decentralized cooperative control, and integrated navigation and localization.

Isaac Kaminer received his Ph.D. degree in Aerospace Engineering Sciences from University of Michigan, Ann Arbor, 1992. He is currently co-director at the Center for Autonomous Vehicle Research, Naval Postgraduate School, Monterey, CA, U.S.A. His research interests include unmanned aerial vehicles, adaptive control, and real-time flight control systems.

Vladimir Dobrokhodov received his Ph.D. degree in Aerospace Engineering Sciences from Zhukovskiy Air Force Engineering Academy, Moscow, 1999. He is currently a research assistant professor at the Center for Autonomous Vehicle Research, Naval Postgraduate School, Monterey, CA, U.S.A. His research interests include guidance, navigation and control of unmanned aerial vehicles, applied nonlinear control, and real-time embedded flight control systems.

Kevin Jones received his Ph.D. degree in Aerospace Engineering Sciences from University of Colorado in May 1993. He is currently a research associate professor at the Center for Autonomous Vehicle Research, Naval Postgraduate School, Monterey, CA, U.S.A. His research interests include micro unmanned aerial vehicles, multidisciplinary design and optimization, fluid mechanics, and aircraft design.