State Estimation of an Autonomous Helicopter Using Kalman Filtering

Myungsoo Jun[†], Stergios I. Roumeliotis[†], Gaurav S. Sukhatme[‡*]

mjun|stergios|gaurav@usc.edu

[†]Department of Electrical Engineering–Systems

[‡]Department of Computer Science
Institute for Robotics and Intelligent Systems
University of Southern California
Los Angeles, CA 90089-0781

Abstract

This paper presents a technique to accurately estimate the state of a robot helicopter using a combination of gyroscopes, accelerometers, inclinometers and GPS. Simulation results of state estimation of the helicopter are presented using Kalman filtering based on sensor modeling. The number of estimated states of helicopter is nine : three attitudes(θ, ϕ, ψ) from the gyroscopes, three accelerations($\ddot{x}, \ddot{y}, \ddot{z}$) and three positions (x, y, z) from the accelerometers. Two Kalman filters were used, one for the gyroscope data and the other for the accelerometer data. Our approach is unique because it explicitly avoids dynamic modeling of the system and allows for an elegant combination of sensor data available at different frequencies. We also describe the larger context in which this work is embedded, namely the design and implementation of an autonomous robot helicopter.

1 Introduction

State estimation is a fundamental need for autonomous robots. The accuracy that we demand from the estimation algorithm however, depends on the control system to be used and the application for which the robot is slated. A large proportion of work in autonomous robotics today deals with ground vehicles and the control systems that allow such vehicles to be autonomous to varying degrees. At the University of Southern California, we have been working on robot helicopters for several years. Figure 1 shows the latest incarnation of the system in flight. As part of a larger

context, our goal is to build an autonomous robot helicopter that can interact with robots on the ground as part of a reconnaissance and surveillance team. The helicopter provides an ideal camera platform for inspection of hazardous material sites, accident sites, crowded urban areas etc.

To date our lab has been working on two aspects of autonomy for such a robot namely low-level control and group behavior. Montgomery et. al. have demonstrated [19] a behavior-based controller comprised of carefully tuned PD control loops that can stabilize a helicopter near hover. Lately, they have extended their work to *learning* a control algorithm for a helicopter [18] using a technique called teaching-by-showing where a human pilot demonstrates control to the robot and the algorithm learns the controller from several input-output pairs of data. The work on heterogeneous group behavior [26] deals with the interaction between the helicopter and the ground robots, and algorithms for cooperative tasking and re-tasking.

We have recently begun an effort to develop algorithms that can accurately estimate the state of a robot helicopter in order to improve the functioning of the control algorithms being developed as part of the work described above. In the paper we report on the first results from this effort. Given the overhead (not to mention the costs associated with crashing) of flight experimentation, we have taken a two staged approach to the problem of state estimation. In the first stage (reported here) we test our algorithm in simulation, using a simulated model of a full-scale helicopter as well as a nonlinear controller that can stabilize it. Based on the encouraging results thus far, we plan to implement our algorithm on the physical system over the next few months. Researchers frequently use

---

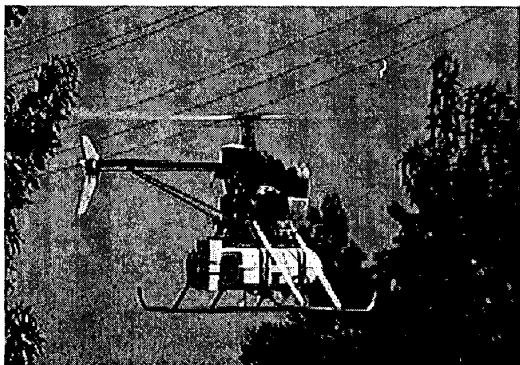*contact author for correspondence

Figure 1: The USC AVATAR: Autonomous Vehicle Aerial Tracking And Retrieval

simulation environments [12] before testing their estimation or control strategies on real helicopters. For example in [22] a simple helicopter model was used for controller design and stability performance analysis. Simulation results for hovering and forward flight conditions were used to illustrate the performance of a tested controller to specific commands.

One method to estimate the state of a helicopter is to use a model. If a state-space helicopter model is available, a Kalman filter based estimator can be built using the model. However the main drawback of this approach is that it is difficult to obtain a good helicopter model. Furthermore, the dynamics of a helicopter are expressed as a set of nonlinear equations. This makes it difficult to construct a Kalman filter. The other method to estimate state is to use a sensor model. The advantage of this is that a complex helicopter model is not needed. Further, Kalman filter need not be rebuilt for each different robot if the sensor suite is unchanged. In this paper, we use the second approach in estimating the state of a helicopter robot.

Kalman filtering [13, 17] is a well known technique for state and parameter estimation. It is a recursive estimation procedure using sequential measurement data sets. Prior knowledge of the state (expressed by the covariance matrix) is improved at each step by taking the prior state estimates and new data for the subsequent state estimation.

Autonomous helicopters have made their appearance with successful results in the mid 90's [25]. Recent work at several centers includes USC [20], [11], Stanford [23], [7], MIT [9], [12] and CMU [2], [6] all

winners of the Association for Unmanned Vehicle Systems International Aerial Robotics competition (1994-1997 respectively).

There are a variety of sensors and architectures that can be used for helicopter state estimation. In one case [3] a two-axis inclinometer, a compass, and a three-axis gyro are considered for attitude estimation. A complementary Kalman filter is used for fusing the data from these sensors. The claim is that the low dynamics of the inclinometers can be effectively compensated when a gyro is used. In the complementary filter the inclinometer dynamics are taken into account. The gyro bias though is ignored. The attitude estimation algorithm has to be combined in a helicopter control algorithm.

Bosse et. al. [5], deal with helicopter velocity estimation during the critical landing phase. An IMU (Inertial Measurement Unit) provides high bandwidth motion estimates from accelerations and rotational velocities that are used for controlling the helicopter. A differential GPS unit provides periodic updates of absolute positions and velocities. The sonar altimeter provides the altitude information for autonomous landings and the compass is utilized to prevent long term drifts in the heading estimate. The authors use an Extended Kalman filter to merge the sensor information into a navigational solution focussed on landing without the help of GPS assuming a flat landing surface.

In [7] the solution to the helicopter state estimation problems comes from using a Carrier Phase Differential GPS (CD-GPS). This was the only sensor for both attitude and position control. With 4 antennas strategically located on-board the helicopter the CD-GPS can be used to determine attitude with accuracy 1-2 degrees. The ground station antenna for the Differential GPS allows a positioning accuracy of 2-3 cm. The capabilities of the system can reach sub-centimeter accuracy. A high bandwidth inner control loop based entirely on the CD-GPS provides attitude stabilization and position control. The main limitation of the system is that commands to fly trajectories or land are possible only if the location of the paths, points or objects can be expressed as GPS coordinates known a priori. A modification to the original system is presented in [23] where a stereo vision system is used for color based identification of objects with a unique color in their environment. Positions are translated to GPS coordinates and an outer control loop allows the helicopter to track objects on the ground.

Various visual tracking approaches have been built to position robot helicopters over fixed targets. These

include [21], where the vision system tracks distinctive features on the ground supported by inertial sensor data. In [4] the authors have developed a vision based system that allows an autonomous helicopter to perform a line tracking task. Vision based obstacle avoidance is described in [27] where a multi-sensor feature-based range-estimation algorithm is proposed for automated helicopter flight. This algorithm can track many features at the same time in multiple image sensors using an Extended Kalman filter to estimate the feature locations in a master sensor coordinate frame. The focus of this paper is on the parallelization of the vision algorithm in order to be used in real time applications.

The rest of this paper is organized as follows. In Section 2, the helicopter model, its characteristics and controller used in the simulation are described; in Section 3, a noise model for the gyroscopes and accelerometers is presented; in Section 4 the architecture of the Kalman filters used in our work are described and in Section 5 the simulation results are given. Section 6 concludes with a summary and a discussion of ongoing and future research.

2   Helicopter Model

There are six rigid body degrees of freedom in the helicopter; three translational (along the three body axes) and three rotational (about these axes). The equations of helicopter dynamics are expressed with respect to the body axis coordinate system. The attitude of the body with respect to the inertial reference frame is defined by Euler angles : θ, ϕ, ψ. The order of rotation is as follows : first, rotate along the X-axis by angle ϕ, then, along the Y-axis by angle θ, and finally along the Z-axis by angle ψ. To each such unique order there exists a corresponding rotation matrix which transforms vectors in the body-centered frame to the inertial frame. The general equations of motion of a helicopter can be found in any standard textbook [10] on the dynamics of flight.

In this work, we use a helicopter model and controller from [16]. The model includes the main rotor rigid body effects, coning and quasi-steady flapping. Although this model, due to [?], contains certain simplifying assumptions, its performance in stability and control studies is fairly well validated. This model has been used extensively in helicopter flight control system studies in the U.K. The controller is designed using Input-Output Linearization[16]. The performance of the controller was evaluated in terms of the *Pilot Handling Quality Requirements* documented in the Aeronautical Design Standard(ADS-33C). By satisfying these specification it is expected that there will be no limitations on flight safety or on the capability to perform intended missions.

In the simulated model, there are four control inputs available to the pilot. These are called collective (θ_o), longitudinal cyclic (θ_{ls}), and lateral cyclic (θ_{lc}), which are control inputs for the main rotor, and tailrotor collective (θ_{ot}), which is for the tail rotor. Four reference commands are given: θ, ϕ, ψ and h.

The outputs of helicopter with commands $\theta = 20^o, \phi = -30^o, \psi = 0$ and $h = 0$ are shown in Figure 2 when there is no noise corruption and no Kalman filter in the loop. This "baseline" figure shows the operation of the controller before we began our experiments. We used these values of command signals throughout the simulation in comparing outputs.

To test the sensor-noise rejection capability of our filters, we injected noise into the baseline simulation (to simulate measurements from real sensors) and added an estimator which reconstructs the state based on these noisy measurements. As we show later, the filters are able to reject fairly realistic levels of noise that may be expected in real flight operations.

3   Noise Models and Characteristics

The AVATAR helicopter robot currently has three gyroscopes (to measure roll, pitch and yaw rates), three accelerometers (to measure, the accelerations in the x, y and z directions), a compass (to measure the yaw angle) and a GPS unit (which provides the x, y, z location of the craft in an inertial coordinate system). The gyroscopes, accelerometers and GPS are part of an integrated avionics unit called the C-MIGITS II Miniature Integrated GPS/INS System from Rockwell. We plan to add two inclinometers to directly measure the roll and pitch as well. The gyroscopes in the avionics unit are Systron Donner Quartz Gyros whose noise profile is well known.

The key difficulty in attitude estimation using gyroscopes is the low frequency noise component, which is also referred as bias or drift that violates the white noise assumption needed for standard Kalman filtering. The noise model used in our simulation is the one in [14, 15]. The model assumes that the gyro noise consists of two elements : rate noise $n_r(t)$ (additive white noise) and a rate random walk $n_w(t)$ (generated when white noise passes through an integrator $1/s$). We used the Systron Donner Quartz Gyro(QRS11-100-420) for numerical values for noise
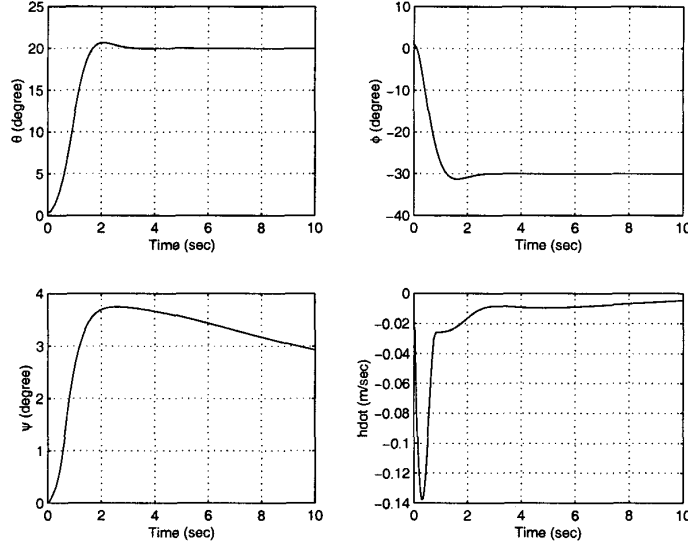
Figure 2: Outputs of helicopter state without noise or Kalman filtering

intensities. The intensities calculated from experiments [1] were $\sigma_r = \sqrt{N_r} = 0.009(deg/sec)/\sqrt{Hz}$ and $\sigma_w = \sqrt{N_w} = 0.0005012(deg/sec)/\sqrt{Hz}$. The level of noise in the compass and inclinometers is assumed to be $\sigma_{ss} = 0.01$. The noise associated with measuring accelerations from the accelerometers and and positions from the GPS is assumed to be white. We used $\sigma_r = 0.07(m)$ for the noise associated with the GPS and $\sigma_q = 0.05(m/sec)$ for the accelerometers.

4 Architectures of Kalman Filters

A Kalman filter model for acceleration is well described in [17]. That approach was followed here for building the filter associated with the accelerometers. However, the Kalman filter model in [17] is one dimensional, that is, it uses just one acceleration measurement. Our simulation uses three components of acceleration, so we use the appropriate rotation matrix and transform the acceleration data with respect to the body reference frame to the acceleration with respect to the inertial reference frame. The rotation matrix can be calculated by using the Euler angles (whose estimates can be obtained from the Kalman filter) for 3-dimensional attitude [8].

For simplicity, we describe a one dimensional Kalman filter here. Let the noise w be a white

Gaussian noise of mean zero and variance kernel $E\{w(t)w(t + \tau)\} = Q\delta(\tau)$ entering at the acceleration level. Let the noise v be a white Gaussian noise of mean zero and variance $E\{v(t)v(t + \tau)\} = R_c\delta(\tau)$ which happens in measuring position r_t, where subscript t denotes a true value. The two error states are defined as

$$\delta r(t) = r_m(t) - r_t(t) \tag{1}$$
$$\delta u(t) = u_m(t) - u_t(t) \tag{2}$$

where u is velocity and subscript m denotes a measured quantity. Then we can write the following state equation:

$$\begin{bmatrix} \dot{r}_i(t) \\ \dot{u}_i(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r_i(t) \\ u_i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} (a_t(t) + w(t)) \tag{3}$$

The state equation of the true position, velocity and acceleration is:

$$\begin{bmatrix} \dot{r}_t(t) \\ \dot{u}_t(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r_t(t) \\ u_t(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} a_t(t) \tag{4}$$

Subtracting Equation 4 from Equation 3, we have

$$\begin{bmatrix} \dot{\delta r}_t(t) \\ \dot{\delta u}_t(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta r_t(t) \\ \delta u_t(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w(t) \tag{5}$$

From these equations, the steady state values can be obtained by solving $\dot{P}(t) = 0$.

$$P = \begin{bmatrix} \sqrt{2}Q^{1/4}R_c^{3/4} & Q^{1/2}R_c^{1/2} \\ Q^{1/2}R_c^{1/2} & \sqrt{2}Q^{3/4}R_c^{1/4} \end{bmatrix} \quad (6)$$

$$K = \begin{bmatrix} \sqrt{2}\omega_n \\ \omega_n^2 \end{bmatrix} \quad (7)$$

where ω_n equals $(Q/R_c)^{1/4}$.

We used four quaternions (also called Euler parameters) in representing the three dimensional attitude. We can easily calculate the rotation matrix from four quaternions [8]. We define $q_{k/k}(\vec{b}_{k/k})$ to be the quaternion (bias) estimate at time t_k based on data up to and including $z(t_k)$, $q_{k/k-1}(\vec{b}_{k/k-1})$ the quaternion (bias) estimate at time t_{k-1} propagated to t_k, right before the measurement update at t_k. The estimated angular velocity is defined as:

$$\vec{\omega}_{k/k-1} = \vec{\omega}_m(t_k) - \vec{b}_{k/k-1} \quad (8)$$

before the update, and as

$$\vec{\omega}_{k/k} = \vec{\omega}_m(t_k) - \vec{b}_{k/k} \quad (9)$$

right after the update.

The full estimated quaternion is propagated over the interval $\Delta t_k = t_k - t_{k-1}$ according to the following equation[28]:

$$q_{k/k-1} = \left(e^{\frac{1}{2}\Omega(\vec{\omega}_{avg})\Delta t_k} + \left(\Omega(\vec{\omega}_{k/k-1})\Omega(\vec{\omega}_{k-1/k-1}) - \right.\right.$$
$$\left.\left. \Omega(\vec{\omega}_{k-1/k-1})\Omega(\vec{\omega}_{k/k-1})\right) \frac{\Delta t_k^2}{48} \right) q_{k-1/k-1} (10)$$

where

$$\Omega(\vec{\omega}) = \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix}, \quad (11)$$

$$\vec{\omega}_{avg} = \frac{\vec{\omega}_{k/k-1}+\vec{\omega}_{k-1/k-1}}{2} \quad (12)$$

The equations for error state covariance propagation, the Kalman gain matrix, the updated covariance and the updated error state are given by:

$$P_{k/k-1} = \Phi(k, k-1)P_{k-1/k-1}\Phi^T(k, k-1) + Q_k (13)$$

$$K_k = P_{k/k-1}H_k^T(H_kP_{k/k-1}H_k^T + R_k)^{-1}, \quad (14)$$

$$P_{k/k} = P_{k/k-1} - K_kH_kP_{k/k-1}, \quad (15)$$

$$\begin{bmatrix} \delta\vec{\theta}_{k/k} \\ \Delta\vec{b}_{k/k} \end{bmatrix} = K_k\Delta z(t_k) \quad (16)$$

where

$$\Phi(k, k-1) = \begin{bmatrix} e^{[[\ \vec{\omega}_{avg}\ ]]\Delta t_k} & -\Psi(\Delta t_k) \\ 0_{3\times3} & I_{3\times3} \end{bmatrix}, (17)$$

$$[[\ \vec{q}\ ]] = \begin{bmatrix} 0 & q_3 & -q_2 \\ -q_3 & 0 & q_1 \\ q_2 & -q_1 & 0 \end{bmatrix}, \quad (18)$$

$$Q_k = \int_0^{\Delta t_k} \begin{bmatrix} \sigma_r^2 I_{3\times3} + \sigma_w^2\Psi(\tau)\Psi^T(\tau) & -\sigma_w^2\Psi(\tau) \\ -\sigma_w^2\Psi^T(\tau) & \sigma_w^2 I_{3\times3} \end{bmatrix} d\tau (19)$$

and

$$\Psi(\Delta t_k) = I_{3\times3}\Delta t_k +$$
$$[[\ \vec{\omega}_{avg}\ ]] (1 - \cos(\omega_{avg}\Delta t_k)/\omega_{avg}^2 +$$
$$[[\ \vec{\omega}_{avg}\ ]]^2 (\omega_{avg}\Delta t_k - \sin(\omega_{avg}\Delta t_k))/\omega_{avg}^3 (20)$$

for detailed equations and derivations the reader is referred to [24].

5 Simulation Results

As mentioned above, nine states were considered and two filters were used to process the data stream from the helicopter's sensors in the simulations reported here. We used $\theta = 20°$, $\phi = -30°$, $\dot{\psi} = 0$ and $\dot{h} = 0$ as reference commands.

Consider Figure 3. This figure shows the values of the real quaternion element q_4, estimated quaternion q_4, measured quaternion q_4 and dead-reckoning estimate of element q_4. As can be seen from the plot, the Kalman filter estimates the real quaternion well from the measured data corrupted by noise. We can also see that the dead-reckoning quaternion is a very good estimate of the real quaternion. The level of noise in angular velocity is very low. Therefore, it is natural that the dead-reckoning quaternion calculated from measured angular velocity is a good estimate of 3-D attitude. However, this is good only for a small time span. Due to the integrating factor (a rate random walk $n_w(t)$), error is accumulated and becomes large after a long time if there is no correction.

Figure 4 shows four error plots; the difference between real position and filtered position data, the difference between real position and dead-reckoning position, the difference between real velocity and filtered velocity, and the difference between real velocity and dead-reckoning velocity. The filtered data for velocity is obtained from adding the value of estimated error state of velocity ($\hat{\delta u}$, see Equation 2) to the value that
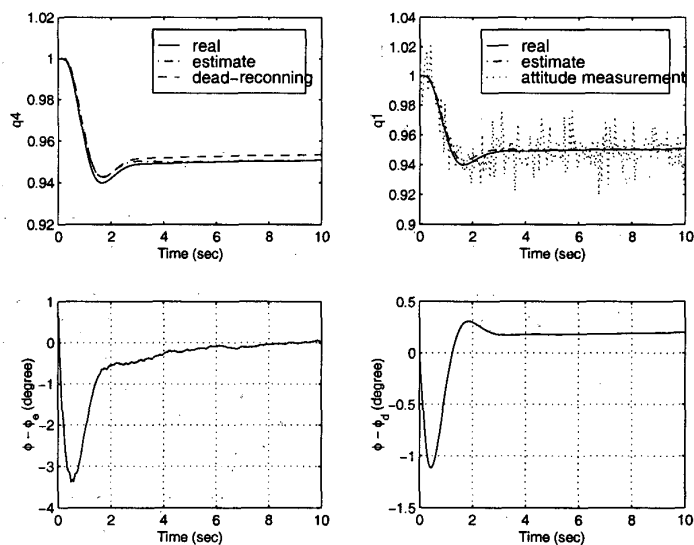
Figure 3: The values of real (solid line), filtered (dash-dotted line), measured (dotted line) and dead-reckoning (dash-dashed line) quaternion element q_4, difference between the real and filtered ϕ (left below), and difference between real and dead-reckoned ϕ (right below)

we get by integrating the measured acceleration. Care must be taken to use the appropriate coordinate system in calculating filtered and dead-reckoned velocity. The estimated attitude(θ_e, ϕ_e, ψ_e) was use to get the rotation matrix in processing filtered velocity and the dead-reckoning attitude(θ_d, ϕ_d, ψ_d) was used to calculate dead-reckoning velocity and position.

The performance improvement can be seen by feeding filtered data to the controller instead of using raw sensor data. The dotted line in Figure 5 denotes the output of the helicopter when raw sensor data was fed to the controller. The solid line is the output when filtered data was used for controller input. As can be seen from the plot, the motion is more stable when the controller used filtered data despite larger overshoot.

6   Conclusions and Future Work

In this paper, a state estimator is developed for a robot helicopter without using a helicopter model. We would like to emphasize that the controller used in the tests here is model based but *not* the estimator. The validity of our approach was verified by feeding filtered data to the controller and observing the output of the helicopter. It remains as future work to verify that our

filters also work well when we change the controller or helicopter without modifying the sensors. After evaluating our Kalman filters for several helicopter models and controllers, our ultimate goal is to implement our algorithm on the real autonomous helicopter in our laboratory.

Acknowledgments

References

[1] R.O. Allen and D. H. Chang. Performance testing of the systron donner quartz gyro. JPL Engineering Memorandum, EM #343-1297, January 5, 1993.

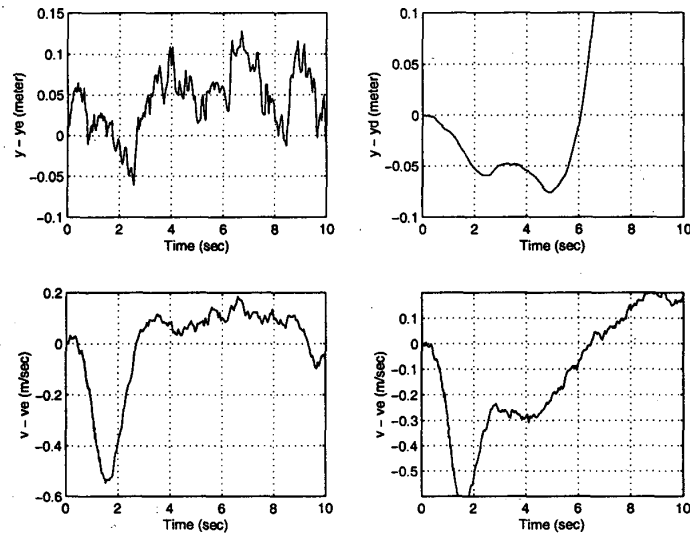[2] O. Amidi, Y. Mesaki, and T. Kanade. Research on an autonomous vision-guided helicopter. In *Conference*

Figure 4: Error in position y when the Kalman filter was used (left above) and when dead-reckoning was used (right above), and error in velocity v when the Kalman filter was used (left below) and dead-reckoning was used (right below)
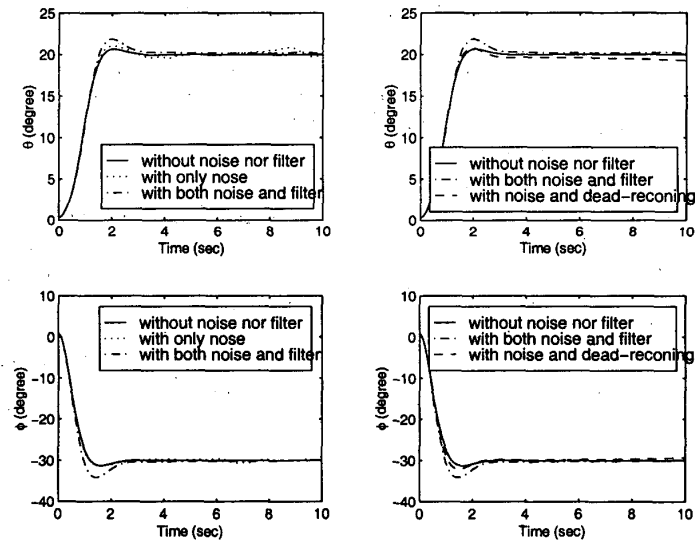

Figure 5: Outputs θ and ϕ of the helicopter when there is no noise nor filter, when there is additive noise and no filter, when we use two Kalman filters in noisy environment, when we use dead-reckoning attitudes without a Kalman filter

on *Intelligent Robotics in Field, Factory, Service and Space*, volume 2, pages 456–463, Linthicum Heights, MD, March 1994.

[3] A.-J. Baerveldt and R. Klang. A low-cost and low-weight attitude estimation system for an autonomous helicopter. In *1997 IEEE International Conference on Intelligent Engineering Systems*, pages 391–395, Budapest, Hungary, Sept 15-17 1997.

[4] P.H. Batavia, M.A. Lewis, and G.A. Bekey. A reduced complexity vision system for autonomous helicopter navigation. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, volume 3, pages 776–780, Nagoya, Japan, May 21-27 1995.

[5] M. Bosse, W. C. Karl, D. Castanon, and P. DeBitetto. A vision augmented navigation system. In *Proceedings of the 1997 IEEE Conference on Intelligent Transportation Systems*, pages 1028–1033, New York, NY, Nov. 9-12 1997.

[6] J. Charles. Cmu's autonomous helicopter explores new territory. *IEEE Intelligent Systems*, 13(5):85–87, Sept-Oct 1998.

[7] A. Conway, P. Montgomery, S. Rock, R. Cannon, and B. Parkinson. A new motion-based algorithm for gps attitude integer resolution. *Journal of the Institute of Navigation*, 43(2):179–190, Summer 1996.

[8] J.J. Craig. *Introduction to Robotics*, chapter 2, pages 55–56. Addison-Wesley, 2nd edition, 1989.

[9] P. DeBitetto, E.N. Johnson, E. Lanzilotta, C. Trott, and M. Bosse. The 1996 mit/ boston university/ draper laboratory autonomous helicopter system. In *AUVSI '96 Proceedings*, pages 911–920, Orlando, FL, July 15-19 1996.

[10] B. Etkin. *Dynamics of Atmospheric Flight*. John Wiley, 1972.

[11] J.F.Montgomery, A.H. Fagg, and G.A. Bekey. The usc afv-i: A behavior-based entry in the 1994 international aerial robotics competition. *IEEE Expert*, 10(2):16–22, April 1995.

[12] E.N. Johnson and P.A. DeBitetto. Modeling and simulation for small autonomous helicopter development. In *AIAA Modeling and Simulation Technologies Conference*, pages 41–51, New Orleans, LA, Aug 11-13 1997.

[13] R. E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 86:35–45, 1960.

[14] A.D. King. Characterization of gyro in run drift. In *Proceedings of the 1984 Symposium on Gyro Technology*, Sept. 1984.

[15] C.R. Kochakian. Time-domain uncertainty charts (green charts): A tool for validating the design of imu/instrument interfaces. Technical Report P-1154, The Charles Stark Draper Laboratory, Aug. 1980.

[16] D. Y. Maharaj. *The Application of Nonlinear Control Theory to Robust Helicopter Flight Control*. PhD thesis, Department of Aeronautics, Imperial College of Science, Technology, and Medicine, 1994.

[17] P. S. Maybeck. *Stochastic Models, Estimation and Control*, volume 141-1 of *Mathematics in Science and Engineering*, chapter 6. Academic Press, 1979.

[18] J. F. Montgomery and G. A. Bekey. Learning helicopter control through 'teaching by showing'. In *1998 IEEE Conference on Decision and Control*, December 1998.

[19] J. F. Montgomery, A. H. Fagg, and G. A. Bekey. The USC AFV-I: A behavior-based entry in the 1994 International Aerial Robotics Competition. *IEEE Expert*, 10(2):16–22, April 1995.

[20] J.F. Montgomery and G.A. Bekey. The usc avatar: A behavior-based entry in the 1995 international aerial robotics competition. In *Proceedings of AUVS '95 International Symposium*, pages 683–691, Washington D.C., July 10-12 1995.

[21] C.H. Oertel. Computer vision sensor for autonomous helicopter hover stabilization. In *Proceedings of the SPIE*, volume 3088, pages 121–129, Orlando, FL, April 21-22 1997.

[22] M.G. Perhinschi and J.V.R. Prasad. A simulation model of an autonomous helicopter. In *Proceedings of RPV/UAV Systems Remotely Piloted Vehicles/Unmanned Airborne Vehicles*, Bristol, UK, March 30 - April 1 1998.

[23] S.M. Rock, E.W. Frew, H. Jones, E.A. LeMaster, and B.R. Woodley. Combined cdgps and vision-based control of a small autonomous helicopter. In *Proceedings of the 1998 American Control Conference*, pages 694–698, Philadelphia, PA, June 24-26 1998.

[24] S. I. Roumeliotis, G. S. Sukhatme, and G. A. Bekey. Smoother-based 3-d attitude estimation for mobile robot localization. Technical Report IRIS-98-363, USC, 1998. http://iris.usc.edu/ irislib/.

[25] M. Sugeno, I. Hirano, S. Nakamura, and S. Kotsu. Development of an intelligent unmanned helicopter. In *Proceedings of 1995 IEEE International Conference on Fuzzy Systems*, volume 5, pages 33–34, Yokohama, Japan, March 1995.

[26] G. S. Sukhatme and J. F. Montgomery. A heterogeneous robot colony for reconnaissance and surveillance. *Autonomous Robots*, Submitted.

[27] R.E. Suorsa and B. Sridhar. A parallel implementation of a multi-sensor feature-based range-estimation method. *IEEE Transactions on Robotics and Automation*, 10(6):755–768, Dec 1994.

[28] J.R. Wertz, editor. *Spacecraft Attitude Determination and Control*, volume 73 of *Astrophysics and Space Science Library*. D. Reidel Publishing Company, Dordrecht, The Netherlands, 1978.