Vision-Based Stabilization for Fixed-Wing Flight

Jeffrey E. Boyd
Department of Computer Science
University of Calgary
boyd@cpsc.ucalgary.ca

Chris Thornton
Department of Computer Science
University of Calgary
chris.thornton@ucalgary.ca

Abstract

Vision and flight are closely linked, leading to a long-standing interest in how the two are connected. Past research has proposed models for vision in control for biological examples of flight, and there has been recent interest in the use of vision for low-level control of small robotic aircraft such as quad-rotor helicopters. In the work presented here, we show a system for stabilization of a small, fixed-wing aircraft in the yaw axis using estimates of parametric optical flow obtained by registration of consecutive video images from a camera mounted on the aircraft. Estimates of angular velocity from the registration replace the values that would otherwise come from a gyro in a conventional stabilization system. No markers or special targets are required – just an environment with enough visual variation to enable image registration. We demonstrate the system in flight and show qualitatively the efficacy of the stabilization from external observation of the aircraft, and from the video acquired from the onboard camera.

1 Introduction

The role of vision in flight is of long-term interest. Whereas much of this interest stems from using an aircraft to carry a vision system payload aloft – leaving the vision system largely independent of the aircraft – we are interested in the coupling of vision with the low-level control of an aircraft. In nature, the connection between vision and control of flight has been of interest [12]. Psychological experiments, show that for humans vision plays an important role in our sense of balance when standing on the ground [13]. Similarly, a human pilot cannot fly at night without the use of instruments in the form of inertial measurement units (IMU). This connection between balance/control and vision leads us to consider vision-based mechanisms for stabilizing flight.

We present a system for stabilization of a fixed-wing aircraft in the yaw axis using estimates of parametric optical flow. The system uses a variation of Mann and Picard's parametric image registration [10] to estimate changes in aircraft orientation, thus acting as a *video gyro*. The control system, shown in Figure 2 is similar to that of Zufferey et al. [18] – whereas Zufferey et al. use a gyro to stabilize the aircraft, we replace the gyro with measurements gleaned from an onboard camera.

2 Background

2.1 Vision in Flight

There is a long interest in the role of vision in flight, some of which we summarize here. Corke et al. [6] provide a review of vision and inertial sensor modalities in both biological and mechanical systems. They highlight the complementary nature of these modalities and their potential for fusion in robotic systems. These properties are applicable to flying as well as ground-based systems.

Neumann and Bülthoff [12] describe a biological vision-based system to control the translatory flight of insects. They show that vision can provide the measurements necessary for control and demonstrate this in simulation. In another biological example, Tucker [16] examines vision in falcons and other birds of prey. Of interest is his proposed *ideal falcon*, a model of typical characteristics of the vision system of falcons and similar birds of prey. Tucker indicates that the falcons have two foveae per eye – one low-acuity fovea 15° off of forward, and one high-acuity fovea approximately 45° off of forward. The movement of the eyeball in the head is negligible, so the falcon must turn either its head or body to change viewing direction. Showing the coupling of vision and control of flight, a falcon is most aerodynamically efficient when its head is facing forward, and therefore uses flight paths that keep the head forward while fixating on prey with a high-acuity fovea.

Many examples of low-level control using vision in flight use quad-rotor helicopters. Most of these examples rely on markers of varying sorts placed in the environment to assist the vision task. For example, Altuğ et al. [1] stabilize a quad-rotor with two cameras – one on ground and one on the aircraft. They acquire visual measurements of the locations of colored blobs, and use this to control flight. Muratet et al. [11] use visual measurements of the divergence of optical flow to compute time-to-impact (the well-known looming effect). They use this to design a controller that avoids collisions while flying in a cluttered environment and demonstrate it in simulation. Tournier et al. [15] developed a specialized target system that uses moiré patterns to assist vision-based measurements. They demonstrate the resulting control system with a quad-rotor in flight. Vision systems have been demonstrated to work in conjunction with inertial systems. For example, Kim and Sukkarieh [9] combine visual and inertial measurements for simultaneous localization and mapping (SLAM) on a fixed wing aircraft. Zhang et al. [17] control a quad-rotor with a combination of vision and an IMU, where the vision system tracks markers placed in the environment. In related work, Kelly and Sukhatme [8] describe a method to calibrate a camera with respect to an IMU for combined visual and inertial control. Finally, Cabecinhas et al. [4] stabilize a quad-rotor with images of landmarks from a pan-tilt-zoom (PTZ) camera on the aircraft and data from other sensors. They actuate the PTZ camera to keep the landmarks in view as the aircraft moves, and demonstrate this in simulation.

Zufferey et al. [18] give an example of vision in fixed-wing flight that is interesting because their aircraft has a mass of only $10g$, which places exceptional constraints on the size and mass of the onboard vision system. Their vision sensor has a single row of pixels to measure divergence in the optical flow and estimate time to impact with walls. As the aircraft approaches a wall, it turns to avoid impact. They project a textured pattern onto the walls to assist the vision system, and stability in flight is achieved by a combination of aircraft configuration and an onboard gyro. Their gyro also feeds forward to the optical flow calculations to remove the rotational component of the flow.

2.2 Fixed-Wing Flight Requirements

Tennekes [14] provides an excellent introduction to the requirements for fixed wing flight. As a first estimate, the relationship between the mass, size, and speed of an airplane (or bird) is

$$\frac{W}{S} \propto V^2, \tag{1}$$

where W is the weight of the aircraft (in N), S is the surface area of the wing (in m^2), and V is the speed (in m/s). The quantity W/S is the *wing loading*. This tells us that if we increase the weight of the airplane, it must fly faster (consuming more power) to stay aloft. This places constraints on our vision system. With small aircraft, we are limited in the mass and power consumption (because power requires batteries with mass) of a vision system.

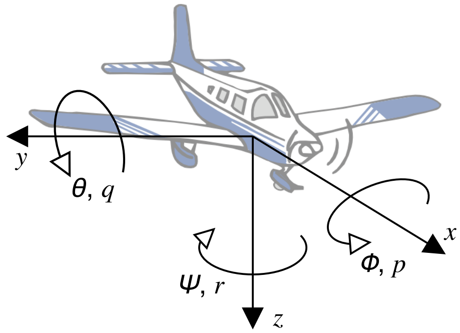


Figure 1. Conventional aircraft coordinate system [5].

Zufferey et al. [18] solve this with custom electronics and working with only one row of pixels to get a total mass of $10g$. Alternatively, one can use a larger airplane such as a mini unmanned air vehicle (UAV). However, UAVs of this size are expensive, require trained operators, and impose a plethora of regulatory and safety constraints. Our choice is to go with smaller a smaller aircraft that is popular with UAV hobbyists, and replace the mass of a vision system with radio-frequency (RF) links to a vision system on the ground. The RF equipment is significantly lighter, making it possible to fly a slower and easier-to-operate airplane, at the cost of limited range and reliability in the RF links.

As previously mentioned, quad-rotors offer an alternative to fixed-wing aircraft and have the advantage of being able to hover. This makes them convenient to operate, especially indoors. However, this comes at a cost of energy efficiency – in the words of Tennekes [14], “Hovering is an uneconomical way of life.” Fixed-wing aircraft make more efficient use of energy, but must maintain airspeed.

3 Rotation from Image Registration

3.1 Coordinate System

For this work, we use the coordinate system conventional in flight [5, 12], shown in Figure 1. The x axis points forward in the direction of flight, y to the right wing, and z is down. Rotation about these axes is denoted by ϕ, θ and ψ, and commonly referred to as *roll*, *pitch*, and *yaw*. Positive ϕ rotates the y axis towards z, positive θ rotates the z axis towards x, and positive ψ rotates the x axis towards y. p, q, and r denote the corresponding angular velocities.

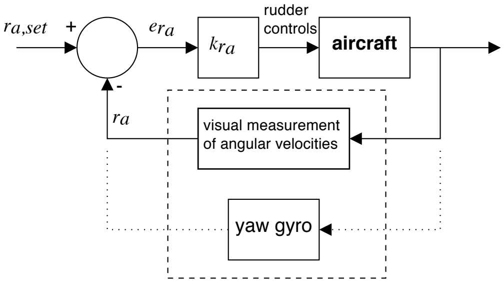


Figure 2. Control system for yaw stabilization, adapted from Zufferey et al. [18]. In our system, computer vision estimation of rotational velocities replaces the gyro.

3.2 Overview

Figure 2 summarizes our system for stabilization. We make the assumption that our aircraft is aerodynamically stable, i.e., with control surfaces in their neutral position and without external perturbation, the aircraft will fly level (or glide if not powered) along a straight path. In this situation, stabilization is only required when the aircraft is perturbed, e.g., as by gusts of wind. In gusty conditions, stabilization is necessary to keep the flight straight and level.

We consider stabilization only in the yaw axis, and leave the problem of pitch stabilization aside. To that end, we start from a yaw stabilization system based on a gyro, e.g., Zufferey et al. [18]. The signal, $r_{a,set}$ sets the desired yaw rate in aircraft coordinates. The gyro senses the yaw of the aircraft, r_a and the control systems corrects by applying rudder to bring the error signal, $e_{r_a} = r_{a,set} - r_a$, to zero. Zufferey et al. use a measurement of time to impact from optical flow to change $r_{a,set}$ for their aircraft to avoid collision with walls. We replace the yaw gyro with three-axis rotational velocity measurements from and onboard video camera and image registration. k_{r_a} is the control-loop proportional gain and is set through experimentation.

3.3 Visual Measurements of Angular Velocity

To measure rotational velocities, we register consecutive pairs of images from an onboard camera to compute Euclidean parameters with three degrees of freedom. The parameters are:

- ϕ (image-plane rotation), and
- t_x and t_y (horizontal and vertical image-plane translations).

ϕ corresponds to ϕ_c, the rotation of the camera about its optical axis. From t_x and t_y, and camera calibration parameters we get θ_c and ψ_c camera rotations. These and the time interval between consecutive frames of video give $\begin{bmatrix} p_c & q_c & r_c \end{bmatrix}^T$, the angular velocity in the camera coordinate frame. Transformation to account for the camera orientation on the aircraft gives $\begin{bmatrix} p_a & q_a & r_a \end{bmatrix}^T$, angular velocity in aircraft coordinates.

To perform the image registration, we modify Mann and Picard's [10] least-squares method for estimating parametric flow to compute Euclidean flow parameters as follows. Start with the Euclidean transformation between coordinates (x, y) and (x', y') in image coordinates,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \quad (2)$$

Assuming ϕ is small, then $\cos\phi \approx 1$ and $\sin\phi \approx \phi$. We can rewrite Eq. (2) as

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & \phi \\ -\phi & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \quad (3)$$

By substitution into the optical flow constraint equation [7],

$$\begin{bmatrix} E_x & E_y \end{bmatrix} \begin{bmatrix} x' - x \\ y' - y \end{bmatrix} = -E_t, \quad (4)$$

where E is an intensity image, subscripts x, y, and t indicate partial derivatives in image space and time, we get

$$\begin{bmatrix} E_x & E_y \end{bmatrix} \left(\begin{bmatrix} 1 & \theta \\ -\theta & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \right) \quad (5)$$
$$= -E_t.$$

We have one instance of Eq. (5) for all pixels in Ω, the overlapping image regions. Rearranging Eq. (5) to solve for $\begin{bmatrix} t_x & t_y & \phi \end{bmatrix}^T$ gives:

$$\begin{bmatrix} E_x & E_y & yE_x - xE_y \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ \phi \end{bmatrix} = -E_t. \quad (6)$$

Using least squares to solve over Ω, gives:

$$A_{3\times3} \begin{bmatrix} t_x \\ t_y \\ \phi \end{bmatrix} = b_{3\times1}, \quad (7)$$

where the elements of A and b are:

$$
\begin{aligned}
a_{11} &= \sum_{\Omega} E_x^2, && (8)\\
a_{12} &= a_{21} = \sum_{\Omega} E_x E_y, && (9)\\
a_{13} &= a_{31} = \sum_{\Omega} E_x (yE_x - xE_y), && (10)\\
a_{22} &= \sum_{\Omega} E_y^2, && (11)\\
a_{23} &= a_{32} = \sum_{\Omega} E_y (yE_x - xE_y), && (12)\\
a_{33} &= \sum_{\Omega} (yE_x - xE_y)^2, && (13)\\
b_1 &= -\sum_{\Omega} E_x E_t, && (14)\\
b_2 &= -\sum_{\Omega} E_y E_t, \text{ and} && (15)\\
b_3 &= -\sum_{\Omega} E_t (yE_x - xE_y). && (16)
\end{aligned}
$$

In the manner described by Mann and Picard, we solve Equation (7) on an image pyramid starting at the coarsest resolution and moving down to the finest, while iteratively refining the solution in each pyramid layer.

Let α_x and α_y be the ratios of the camera focal length to the x and y pixel size obtained by calibrating the camera. Then we have

$$
\begin{aligned}
\phi_c &= \phi, && (17)\\
\theta_c &= \arctan\left(-t_y/\alpha_y\right), \text{ and} && (18)\\
\psi_c &= \arctan\left(t_x/\alpha_x\right), && (19)
\end{aligned}
$$

where ϕ_c, θ_c, and ψ_c are rotations in the camera-centered coordinate system. Let Δt be the time interval between frames E and E'. The camera-centered angular velocity estimates are:

$$
\begin{aligned}
p_c &= \phi_c/\Delta t, && (20)\\
q_c &= \theta_c/\Delta t, \text{ and} && (21)\\
r_c &= \psi_c/\Delta t. && (22)
\end{aligned}
$$

Let ${}^aR_c$ be the orthogonal matrix that transforms camera-centered to aircraft-centered coordinates. We get:

$$
\begin{bmatrix} p_a & q_a & r_a \end{bmatrix}^T = {}^aR_c \begin{bmatrix} p_c & q_c & r_c \end{bmatrix}^T, \qquad (23)
$$

thus giving the required angular velocities of the aircraft.

4 Test System

4.1 Overview

Figure 3 shows the system we use to test our yaw stabilization system. The system has two sides: airborne and ground. Three radio frequency (RF) links connect components on the two sides. These perform the following functions.

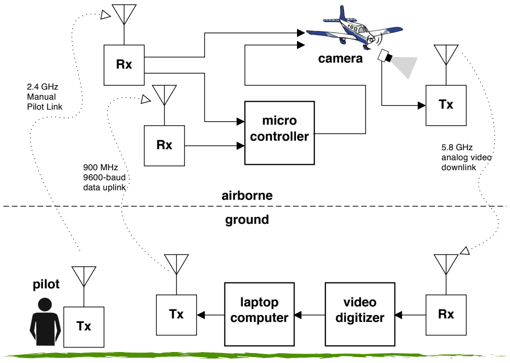


Figure 3. Schematic diagram of aircraft stabilization test system.

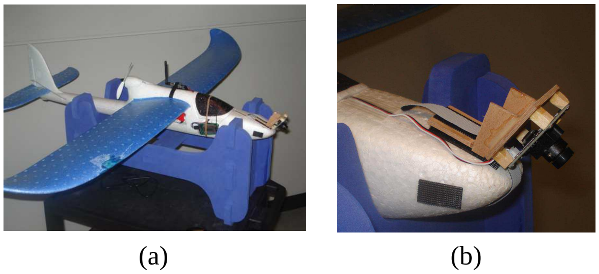
1. A 5.8*GHz* link transmits analog video data from the aircraft to the ground station.

2. A 900*MHz* link transmits digital data from the ground to the aircraft. Although this is a bi-directional link, we use it as an uplink only to maximize the bit rate at 9600 baud.

3. A 2.4*GHz* link provides manual control to a trained radio-control pilot on the ground.

The control system of Figure 2 is embedded in the test system as follows. A camera mounted on the nose of the aircraft acquires video images during flight. The 5.8*GHz* link transmits these images to the ground systems, where they are digitized for processing on a laptop computer. The laptop computer computes the angular velocities of the aircraft using the method described in Section 3 and transmits $\begin{bmatrix} p_a & q_a & r_a \end{bmatrix}^T$ from Equation (23) to the aircraft by the 900*MHz* uplink. Onboard the aircraft, a microcontroller receives the angular velocity data and uses it to compute e_{r_a}, apply the gain, k_{r_a}, and actuate the rudder to steer the aircraft and camera.

The microcontroller is an *Arduino* [3] with an Atmel ATMega328 in the form of an *ArduPilot* [2], chosen because it is small, light, and has low power consumption making it excellent for small airborne applications. Furthermore, because the ArduPilot was designed specifically for UAV experimentation, it facilitates the use of conventional aircraft servo motors and includes a failsafe mechanism to return manual control to the ground whenever necessary (for safety and equipment longevity). The ArduPilot is not suitable for real-time video computations, but is more than adequate for the other computations in our control system.



(a) (b)

Figure 4. Unmanned aircraft (Multiplex Easystar) configured for stabilization tests: (a) full aircraft with camera and radio equipment visible, and (b) close-up view of camera fixed in forward facing direction for stabilization.

The aircraft, shown in Figure 4(a), is a Multiplex Easystar. The Easystar is a popular platform for experimentation with small UAVs. It has excellent aerodynamic stability and has sufficient wing area to lift a small camera and transmitter at approximately $10m/s$ airspeed. Fully configured for our tests, the aircraft has a mass of 830*g*, which is 150*g* over the manufacturers specification. A *pusher propeller* configuration means there is no propeller at the front of the aircraft that can interfere with the camera. The aircraft does not have ailerons for roll control, using rudder (yaw), elevator (pitch) and throttle (thrust) control only. However, when the rudder is actuated, the aerodynamic properties of the aircraft will induce roll and yaw simultaneously.

We mount the camera facing forward and down (approximately $40°$), as shown in Figure 4(b). This ensures that the camera will normally see textured regions on the ground – better for image registration. Depending on altitude, optical flow due to the ground passing under the aircraft can significantly affect the angular velocities measured. However, when facing forward, the yaw measurement is not affected.

4.2 Method

We test our system with a progression of experiments. This starts with mounting the system on a ground robot in the laboratory. While the results of this are trivial, it is a necessary step to verify the system without catastrophic results.

The next step is to fly the aircraft with a yaw gyro onboard, while carrying the camera. This serves two purposes. First, we verify the range of the 900*MHz* and 5.8*GHz* links before they become part of a critical control loop (we know the quality and range of the 2.4*GHz* radio-control link from prior extensive use). Second, we can verify the control loop running on the microcontroller with known hardware and establish a suitable range of values for k_{r_a}. Using the 2.4*GHz* link, the pilot can set k_{r_a} manually until the aircraft flies in straight in gusts without intervention. Note that gyro control modules are available off-the-shelf that provide exactly this function, but using our own system with a gyro gives us numerical values for k_{r_a} that we can then transfer to the vision system.

Finally, we fly the aircraft with the vision system in the loop and the gyro removed. During takeoff, the pilot uses the failsafe system to have complete control of the aircraft. Once the aircraft is at a safe altitude, the pilot engages the vision control loop, and adjusts k_{r_a} (from a suitable range determined by the gyro experiments) until the aircraft flies straight in gusts without manual intervention.

5 Observations

We flew the aircraft on a day with south-south-east winds averaging approximately $15km/h$ with gusts up to $20km/h$ and lulls down to $7km/h$. Figure 5 shows video images from the aircraft while flying with video stabilization. The flight path is approximately south over a runway with a heading of $200°$ – so the wind is coming from the upper-left corner of the images.

Without the control loop engaged, the video images show the constant movement of the aircraft as the wind perturbs it and the pilot makes continual adjustments to stay on course. While it was not our goal to stabilize the video for human viewers, the airborne video without control, subject to all of the aircraft motion, was decidedly unpleasant to view and difficult to interpret.

With the control loop engaged, and no pilot intervention, the airborne video gives a stable view without the rapid movements characteristic of the un-stabilized flight. Sample video shows the aircraft flying for periods of up to $30s$ without pilot intervention. Intermittent gusts perturb the aircraft but it always quickly returns to straight flight. Longer durations were not possible because pilot intervention was necessary to keep the aircraft within range of the RF links, and away from nearby no-fly areas.

The efficacy of stabilization is easy to see from the ground as the gain, k_{r_a}, varies. With low gain, there is no feedback and the aircraft is unable to keep a heading through the wind. As the gain increases, the pilot can release control to the aircraft. As the gain increases further, the tail of aircraft moves visibly as the control loop overcorrects the yaw.

Our evaluation is subjective – we see the aircraft in stable flight – but it would be preferable to have empirical evidence. Ideally, we would like to have compared readings from a physical gyro in flight and compare measurements when video stabilization is on or off. Unfortunately, our use of RF links with limited bandwidth to save mass in the airplane prevents this. The 900*MHz* datalink bandwidth drops precipitously when used for bidirectional communication,

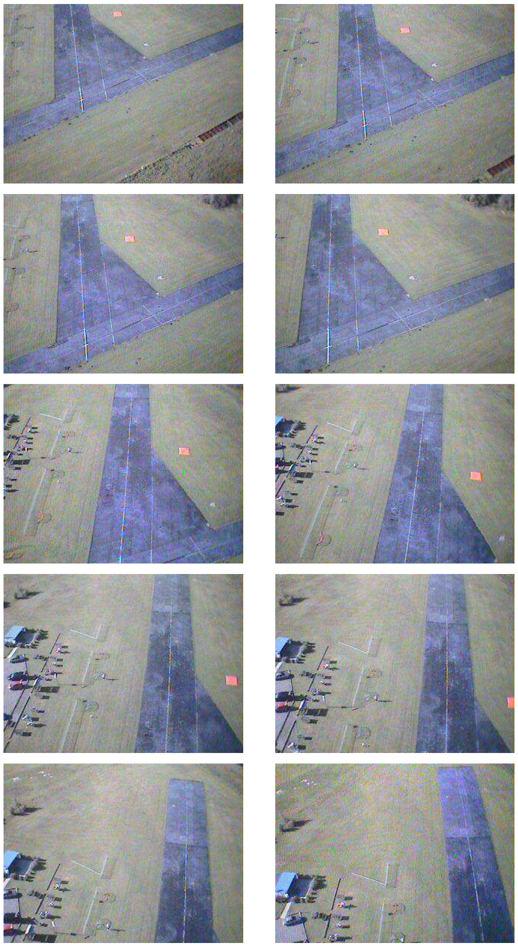


Figure 5. Video frames extracted at equal intervals from $13s$ of video-stabilized flight. Order is row-major.

so acquiring telemetry data was not possible.

6 Discussion

A forward-facing camera allowed us to ignore visual yaw induced by viewing the ground. This differs from Tucker's ideal falcon model [16] in which the foveae in each eye are 15° and 45° off center. Thus, if a falcon were to use this method to assist stabilizing flight, it would need to account for ground speed. One option based on naive speculation is to use flow from eyes on opposite sides of the head so that visual yaw induced by ground speed cancels between the two.

We did not implement pitch control. While we do get visual pitch measurements, two factors confound its use in stabilization. First, while drift in yaw control results in a gradual turn of the aircraft, drift in pitch leads to a stall or a dive. Intertial measurements from an accelerometer are necessary. Second, ground speed produces visual pitch that must be accounted for in control, and again, this would likely require an accelerometer.

The week link in our system is the RF communication. We have limited range, and even when in range, we get frequent dropped, incomplete, or distorted video images. These images lead to erroneous registrations and bad data in the feedback loop. Fortunately, these are only intermittent most of the time, and the aerodynamic properties of the aircraft allow it to recover from momentary deflections of the rudder.

We chose to compute Euclidean parameters for image registration. It is equally possible to use affine parameters decomposed into rotation, scale and translational degrees of freedom.

7 Conclusion

We have demonstrated a working system that uses visual image registration to stabilize fixed wing flight in the yaw axis. Significantly, the systems requires no special markers or targets, only enough visual variation in its environment to facilitate image registration. For real-time operation on a small aircraft, current technology requires that the vision processing must be done on the ground. Lighter, low-power vision systems will be necessary to duplicate the process onboard an aircraft. of this size.

References

[1] E. Altuğ, J. P. Ostrowski, and C. J. Taylor. Control of a quadrotor helicopter using dual camera visual feedback. *International Journal of Robotics Research*, 24(5):329–341, May 2005.

[2] C. Anderson. Ardupilot main page. Retrieved September 6, 2011, from http://diydrones.com/profiles/blogs/ardupilot-main-page, January 2009.

[3] Arduino home page. Retrieved September 7, 2011, from http://www.arduino.cc/.

[4] D. Cabecinhas, C. Silvestre, and R. Cunha. Vision-based quadrotor stabilization using a pan and tilt camera. In *IEEE Conferenc on Decision and Control*, pages 1644–1649, Atlanta, GA, December 2010.

[5] M. V. Cook. *Flight Dynamics Principles*. Elsevier, Amsterdam, 2nd edition, 2007.

[6] P. Corke, J. Lobo, and J. Dias. An introduction to inertial and visual sension. *International Journal of Robotics Research*, 26(6):519–535, June 2007.

[7] B. K. P. Horn. *Robot Vision*. MIT Press, Cambridge, Massachusetts, 1986.

[8] J. Kelly and G. S. Sukhatme. Fast relative pose calibration for visual and inertial sensors. In *11th International Symposium on Experimental Robotics*, Athens, Greece, July 2008.

[9] J. Kim and S. Sukkarieh. Real-time implementation of airborne inertial slam. *Robotics and Autonomous Systems*, 55(3):62–71, January 2007.

[10] S. Mann and R. W. Picard. Video orbits of the projective group: a simple approach to featureless estimation of parameters. *IEEE Transactions on Image Processing*, 6(9), September 1997.

[11] L. Muratet, S. Condieux, Y. Brierre, and J.-A. Meyer. A contribution to vision-based autonomous helicopter flight in urban environments. *Robotics and Autonomous Systems*, 50(4):195–209, March 2005.

[12] T. R. Neumann and H. H. Bülthoff. Insect inspired visual control of translatory flight. In *6th European Conference on Advances in Artificial Life*, volume LNAI 2159, pages 627–636, 2001.

[13] S. E. Palmer. *Vision Science Photons to Phenomenology*. MIT Press, Cambridge, MA, 1999.

[14] H. Tennekes. *The Simple Science of Flight*. MIT Press, Cambridge, MA, 1992.

[15] G. P. Tournier, M. Valenti, J. P. How, and E. Feron. Estimation and control of a quadrotor vehicle using monocular vision and moiré patterns. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, Keystone, CO, August 2006.

[16] V. A. Tucker. Deep fovea, sideways vision and spiral flight paths in raptors. *Journal of Experimental Biology*, 203(24):3745–3754, 2000.

[17] T. Zhang, Y. Kang, M. Achtelik, K. Kühnlenz, and M. Buss. Autonomous hovering of a vision/imu guided quadrotor. In *International Conference on Mechatronics and Automation*, pages 2870–2875, Changchun, China, August 2009.

[18] J.-C. Zufferey, A. Klaptocz, A. Beyeler, J.-D. Nicoud, and D. Floreano. A 10-gram microflyer for vision-based indoor navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3267–3272, Beijing, China, October 2006.