



Faculty Publications

2007-02-01

Performance Evaluation of Vision-Based Navigation and Landing on a Rotorcraft Unmanned Aerial Vehicle

David Hubbard

Timothy W. McLain
mclain@byu.edu

Bryan S. Morse
morse@byu.edu

Colin Theodore

Mark Tischler

Follow this and additional works at: <https://scholarsarchive.byu.edu/facpub>



Part of the [Computer Sciences Commons](#)

Original Publication Citation

D. Hubbard, B. Morse, C. Theodore, M. Tischler, and T. McLain, "Performance evaluation of vision-based navigation and landing on a rotorcraft unmanned aerial vehicle," in WACV '7: Proceedings of the Eighth IEEE Workshop on Applications of Computer Vision, IEEE Computer Society, February 27.

BYU ScholarsArchive Citation

Hubbard, David; McLain, Timothy W.; Morse, Bryan S.; Theodore, Colin; and Tischler, Mark, "Performance Evaluation of Vision-Based Navigation and Landing on a Rotorcraft Unmanned Aerial Vehicle" (2007). *Faculty Publications*. 274.
<https://scholarsarchive.byu.edu/facpub/274>

This Peer-Reviewed Article is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Faculty Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact ellen_amatangelo@byu.edu.

Performance Evaluation of Vision-Based Navigation and Landing on a Rotorcraft Unmanned Aerial Vehicle

David Hubbard¹, Bryan Morse¹, Colin Theodore², Mark Tischler³, Timothy McLain⁴

¹Department of Computer Science
Brigham Young University
Provo, UT 84602

² San Jose State Foundation
Ames Research Center
Moffett Field, CA 94035

³US Army Aeroflightdynamics Directorate
Ames Research Center
Moffett Field, CA 94035

⁴Department of Mechanical Engineering
Brigham Young University
Provo, UT 84602

Abstract

A Rotorcraft UAV provides an ideal experimental platform for vision-based navigation. This paper describes the flight tests of the US Army PALACE project, which implements Moravec's pseudo-normalized correlation tracking algorithm. The tracker uses the movement of the landing site in the camera, a laser range, and the aircraft attitude from an IMU to estimate the relative motion of the UAV. The position estimate functions as a GPS equivalent to enable the rotorcraft to maneuver without the aid of GPS. With GPS data as a baseline, tests were performed in simulation and in flight that measure the accuracy of the position estimation.

1. Introduction

Unmanned Aerial Vehicles (UAVs) offer significant advantages in many environments classified as dull, distant, or dangerous. In particular, the use of computer vision as an alternative to GPS extends the envelope of potential missions to include low-altitude operation and landing. GPS ranks as one of the most commonly used data sources for robotic applications. However, a UAV may lose GPS reception at a critical time while landing in urban canyons, hostile military theaters, or rugged terrain.

A fully autonomous UAV landing poses two problems. First, the UAV must determine a safe landing site. Second, the UAV must land at the site robustly. This second problem is the focus of this paper. Rotorcraft UAVs (R-UAVs) and other vertical takeoff or landing (VTOL) craft have an advantage over fixed-wing UAVs when landing at

an unprepared site. An R-UAV only requires a smooth landing site approximately as large as itself. An R-UAV with autonomous take off and landing capability can automate many important but repetitive tasks. In a military setting, the ability to land autonomously can facilitate deploying aircraft to a forward location, even if GPS jamming is encountered. Wilderness and disaster search and recovery patrols also potentially benefit from a landing capability, which could perform dangerous rescue operations involving landing at an unprepared site and where, due to rugged terrain, GPS reception may be blocked during landing. An R-UAV is also especially useful in urban terrain, where space is at a premium and ground robots cannot travel as rapidly. Remote scientific work, such as on Mars, is another application for autonomous landing at an unprepared site without the aid of GPS. Such scenarios motivate the use of vision-based navigation on a rotorcraft UAV to automate the landing process.

1.1. Project Overview

The Precision Autonomous Landing Adaptive Control Experiment (PALACE), a US Army Technology Objective (ATO) carried out by the US Army Aeroflightdynamics Directorate (AFDD) at Ames Research Center, develops and tests autonomous landing technology on a Yamaha RMAX R-UAV. The landing process begins at 30 m above ground level (AGL) and proceeds down a -60° slope (see Figure 1). The R-UAV employs a hybrid approach, initially navigating by GPS and transitioning to vision-based navigation at 12 m AGL. As the R-UAV reaches 2 m AGL and approaches the limits of the vision system, it transitions to inertial navigation for the final landing steps [13]. In addi-

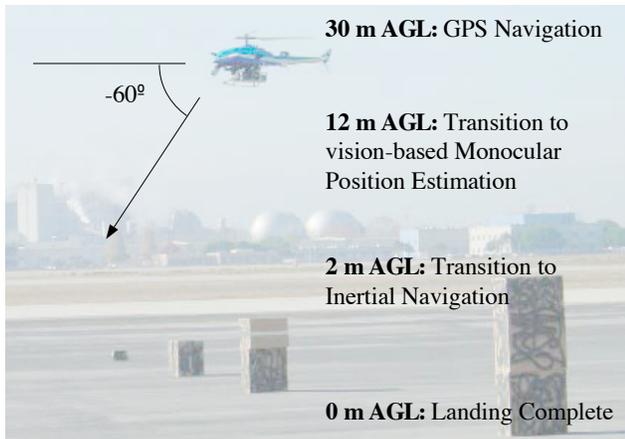


Figure 1. Overview of PALACE landing

tion, during the stage when the vision-based navigation system is active, the R-UAV employs multiple sensory inputs: the monocular camera, a single laser range aimed at -60° to measure the range to the center of the camera image, and the inertial measurement unit (IMU) data to estimate the aircraft attitude. A separate algorithm, not discussed in this paper, performs safe landing area determination (SLAD) using stereo ranging during the landing process [13]. The SLAD algorithm generates a range map of the landing site with increasing precision as the R-UAV descends. The tracking algorithm uses the landing site automatically generated from the SLAD algorithm.

1.2. Related Work

Several pilotless landing techniques have demonstrated excellent reliability, including the use of millimeter-wave radar, monocular tracking, and passive stereo ranging. One of the first autonomous UAV landing demonstrations occurred in 2002, when the Sierra Nevada Corporation landed a Hunter UAV using the UAV Common Automatic Recovery System (UCARS). They produce several military-grade millimeter-wave radar solutions which use a beacon at the landing site to guide the UAV [8]. The Jet Propulsion Laboratory (JPL) research in autonomous landings includes a robust algorithm using monocular vision to land at a site designated with an “H” symbol [9]. JPL has employed a similar method to land on a moving target [10]. They also have demonstrated stereo reconstruction for landing site selection [2]. The University of California Berkeley operates a UAV that estimates motion by solving the system of equations that relate the movement of a prepared target in multiple camera images, and lands on the target without the aid of GPS [11]. The Georgia Institute of Technology operates an R-UAV using contour-based tracking [3]. Relevant publications regarding the PALACE project introduce the machine

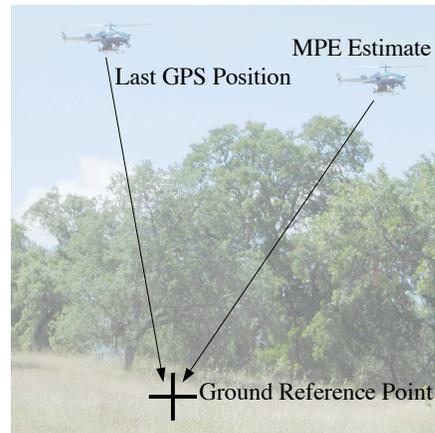


Figure 2. Monocular Position Estimation

MPE Algorithm:

1. Estimate Ground Reference Point relative to GPS.
2. Every 10 Hz, estimate new UAV position based on perceived motion of Ground Reference Point.

vision algorithms employed [4], the full-system simulation and validation environment [12], and the flight tests [13].

2. PALACE System Architecture

The PALACE project uses a modified RMAX operated by the NASA/Army Autonomous Rotorcraft Project (ARP) [14] includes avionics and sensors that enable vision-based navigation, including multiple CPUs for navigation and experimentation. The PALACE software runs on a Pentium-III 700 MHz processor, and interfaces directly with a Firewire Point Grey Flea camera that captures 640×480 grayscale images and a SiCK PLS-300 to provide a laser return at -60° . GPS and IMU data are read from another processor.

The PALACE software aims to act as a drop-in replacement for GPS for the terminal landing phase of autonomous flight. The computer vision algorithms run at 10 Hz, like the GPS receiver. Once the vision-based navigation is initialized with the selected landing site, the GPS signal is only recorded for validation. The camera field of view necessitates switching to inertial navigation from the IMU data at 2 m AGL. Normal aircraft motion below this altitude could induce image motion that exceeds the camera field of view. To complete the landing, the UAV continues to descend until weight-on-wheels switches sense contact with the ground.

2.1. Monocular Position Estimation

The Monocular Position Estimation (MPE) algorithm is initialized with the current aircraft GPS coordinates. Then, given the position of the landing site in the camera image, the MPE algorithm can compute the coordinates of the landing site. This ground reference point then becomes the basis for estimating the aircraft’s position (see Figure 2). Motion

in the camera image implies aircraft motion, provided the ground reference point remains fixed. To resolve three unknowns (the aircraft position) from a two-dimensional camera image, the MPE algorithm relies on a single laser measurement of the distance to the image plane. Assuming the ground is flat within the camera's field of view, this will be the distance to the ground reference point.

The MPE algorithm relies on the Förstner interest operator [1] and pseudo-normalized correlation algorithm [7] implemented and tested at JPL [5]. The MPE algorithm prepares to track the landing site using the Förstner interest operator to identify a point where Moravec's pseudo-normalized correlation algorithm will find sufficient texture to successfully track image motion. Then, the pseudo-normalized correlation can iterate over each camera image, tracking the motion of the landing site, by searching for a small template from the first image in the next image. Successful tracking depends on the global maximum of the correlation function occurring at the new image of the landing site. Moravec derives the pseudo-normalized correlation to give an accurate maximum where the image matches the template without computing a square root. This approximation makes the algorithm usable on systems with limited computing power.

Normalized cross-correlation [6] compares a small $N \times N$ -pixel template image I_b to an $N \times N$ region of a larger image I_a . The comparison is performed over an $M \times M$ search window to find the highest correlation score. Given the mean and variance of the template, $\mu_b = \frac{\sum_{u,v} I_b(u,v)}{N^2}$ and $\sigma_b^2 = \frac{\sum_{u,v} [I_b(u,v)^2]}{N^2} - \mu_b^2$, and the mean and variance of the $N \times N$ region of I_a at (x, y) , $\mu_a(x, y) = \frac{\sum_{u,v} I_a(x+u, y+v)}{N^2}$ and $\sigma_a^2(x, y) = \frac{\sum_{u,v} [I_a(x+u, y+v)^2]}{N^2} - \mu_a^2$, the normalized cross-correlation is

$$\lambda(x, y) = \frac{\sum_{u,v} [I_a(x+u, y+v) - \mu_a(x, y)][I_b(u, v) - \mu_b]}{\sqrt{\sigma_a^2(x, y)\sigma_b^2}}$$

The pseudo-normalized correlation approximates this by dropping the $I_a(x+u, y+v)\mu_b$ and $I_b(u, v)\mu_a(x, y)$ terms and using the arithmetic mean instead of the geometric mean in the denominator:

$$c(x, y) = \frac{\sum_{u,v} [I_a(x+u, y+v)I_b(u, v)] - N^2\mu_a(x, y)\mu_b}{N^2 \frac{1}{2} [\sigma_a^2(x, y) + \sigma_b^2]}$$

The MPE algorithm selects the pixel (x, y) with the highest correlation score and saves a new template $I_b = I_a(x, y)$ for the next iteration. It can now estimate the UAV's position relative to the ground reference point (X_0, Y_0, Z_0) using the distance measured by the laser range finder, R , and the 3×3 rotation matrix from the IMU, M :

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} + M \begin{pmatrix} \frac{R}{f}x \\ \frac{R}{f}y \\ R \end{pmatrix} \quad (1)$$

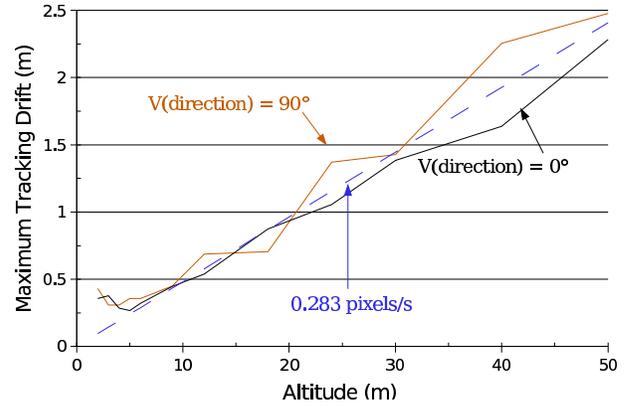


Figure 3. Simulation shows 0.283 pixels/s maximum drift in a 120 s flight. Tests with wind head-on to the R-UAV (0°) and sideways (90°) show that the tracking algorithm operates correctly in turbulence.

where f is the focal length of an equivalent pinhole camera.

3. Performance Evaluation

A performance evaluation of the PALACE software in simulation and flight shows sufficient accuracy to operate as a GPS replacement. In order to use the MPE algorithm in a closed-loop system, the position estimate must have reasonably low noise and low latency. The position controller can tolerate up to 400 ms of delay for small inputs, but the attitude controller is much more sensitive. The MPE algorithm uses camera, laser, and IMU data simultaneously, which implies that error induced by mismatched input timing could result in aircraft attitude feedback as position estimation error, which will be fed indirectly to the attitude controller with greatly increased delay. This causes oscillations and instability.

By measuring each sensor using frequency sweeps and calibrating the inputs, the MPE algorithm can receive synchronous data for its inputs. Experimentally, the camera and laser measurements were found to be close to instantaneous. (They are connected directly to the CPU running the MPE algorithm.) The IMU data had a consistent 44 ms delay, and by adding a compensator in software the closed-loop system performed as desired. Eq. 1 has no inherent delay, so only signal processing delays limit the system's performance.

After testing the system stability, a comparison between GPS and MPE accuracy highlights sources of error in the vision system. The iterative nature of the tracking algorithm compounds the inherent rounding due to the finite camera

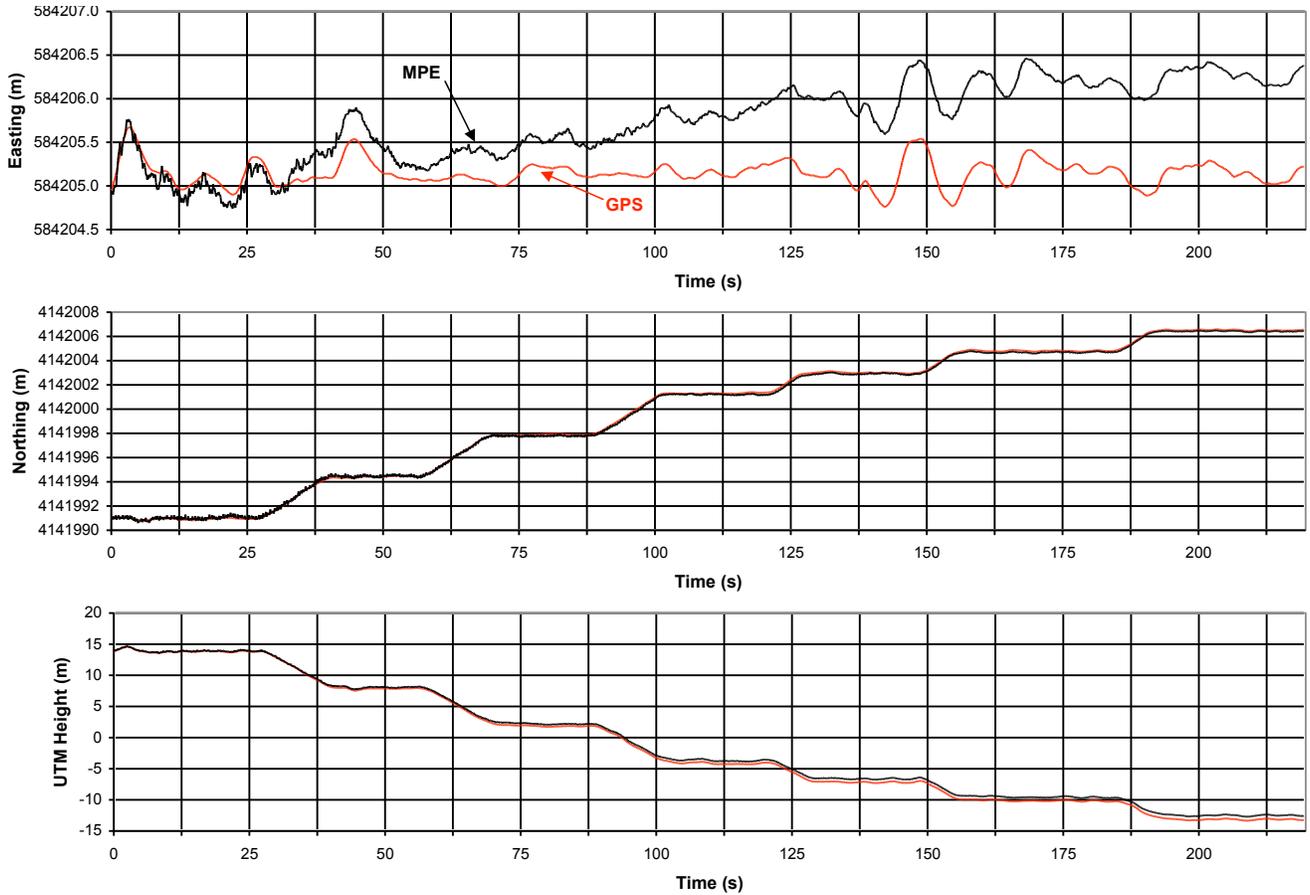


Figure 4. Flight test over grass demonstrates both zero-mean sensor noise, most visible in easting at high altitude, and tracking drift, which accumulates slowly with time. The data shows the R-UAV descending in 6 m increments, and pausing to evaluate performance at each altitude. MPE provides good accuracy within the finite time span for an autonomous landing.

resolution. This leads to cumulative error in the position estimate over time, which manifests as “drift” in the estimated position. The PALACE software operates well in daylight conditions and over naturally textured surfaces, such as concrete, asphalt, and grass, where the tracking algorithm can find a good match for the template from frame to frame. Within these conditions, simulation and flight testing show a small cumulative drift and zero-mean noise proportional to altitude.

3.1. Simulation

Simulation testing of tracker drift suggests the hypothesis that assuming proper conditions for tracking (i.e. sufficient light and surface texture), an expected maximum drift rate is computable [12]. The tracking algorithm rounds the true target location in the image x_T to the integer approximation $x_e = \lfloor x_T \rfloor$, and x_T is equally likely to lie anywhere

in $[x_e, x_e + 1)$. The linear transformation in Eq. 1 maps this to a position uniformly distributed over $[X_e, X_e + \frac{R}{f})$. The two dimensions of rounding error in x and y may appear in X, Y and Z by means of the rotation matrix M .

This result is observable in simulation. Zero-mean noise caused by the finite accuracy of the imaging sensor increases with altitude. Similarly, the drift rate of the tracking algorithm, measured in pixels per second, is proportional to $\frac{R}{f}$ when measured in meters. Simulations run to determine the maximum drift rate (see Figure 3) are performed by running the tracking algorithm for 120 s on simulated imagery to find the maximum difference between MPE and GPS with wind head-on and sideways. The results are then averaged at each altitude point. Wind direction, with the resulting increase in turbulence, was found not to effect the tracker significantly. Simulation results measure the maximum drift at 34 pixels for 120 s, or 0.283 pixels/s [12].

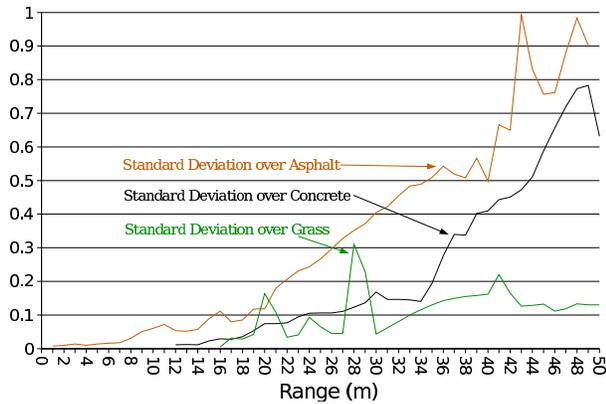


Figure 5. The laser range measurement directly affects position estimation error. The SiCK PLS-300 has excellent standard deviation of measurement error over flat surfaces such as asphalt, concrete, and grass.

3.2. Flight

Flight tests to demonstrate successful autonomous landings at an unprepared site [13] show a similar drift rate to simulation results. Figure 4 demonstrates a descent, where the MPE estimate diverges slowly from the GPS reference over the course of almost 4 minutes. Two types of error appear in the plots, especially visible in the Easting plot: zero-mean noise and drift accumulated in the tracking algorithm. For example, the laser range finder has excellent noise characteristics, with a linear increase in noise as altitude increases (see Figure 5). This is more visible as zero-mean noise in (X, Y, Z) when R is larger.

Performance evaluations over concrete, asphalt, and grass for 120 s indicate that the tracking algorithm does as well as in simulation, drifting at most only 22.8 pixels, or 0.19 pixels/s (see Figure 6). 17 successful landings with various obstacles and surface textures have shown a 1 m or better accuracy at the landing site. Important lessons on the applicability of correlation-based tracking algorithms are that the algorithms perform well enough to be useful as a replacement for GPS, that the hardware used performs well in daylight and overcast conditions, but indoor or nighttime conditions do not provide enough light, and that a hybrid approach using vision down to about 2 m AGL and inertial navigation for the final landing resolved issues related to the limited field of view at low altitudes. The vision system performs well in tests up to 30 m AGL, though the project only called for operation at 12 m AGL. The PALACE project successfully achieved the goal of autonomous landing capability.

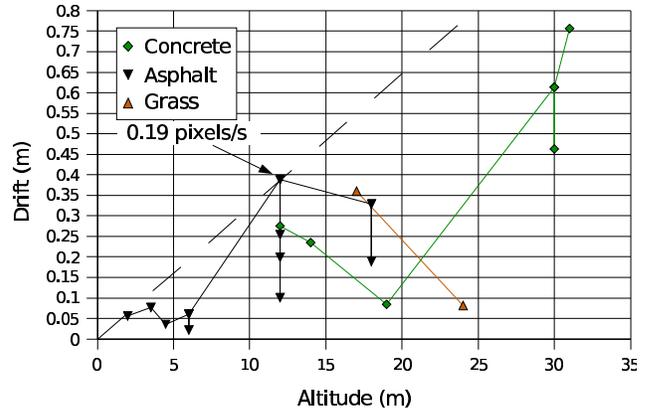


Figure 6. Hover flights for 120 s over concrete, asphalt, and grass show a maximum drift of 0.19 pixels/s.

4. Summary and Conclusions

The full autonomous landing has been successfully demonstrated numerous times using the MPE algorithm. The ARP RMAX provides an ideal platform for developing robust vision-based navigation and landing systems. Autonomous landings without the aid of GPS advance the state of the art for UAV operations, and synthesize several important technologies in a useful application. By using both computer vision and a single laser measurement, the MPE algorithm can estimate position as accurately as a GPS sensor.

Several important areas that future research could address include the addition of a night vision camera or new CMOS cameras for low light conditions, improvements to the MPE and tracking algorithms' accuracy with a corresponding performance evaluation, synthesis of multiple tracking methods to better handle poorly textured landing surfaces, and other possible applications such as visual odometry or self localization and mapping.

Acknowledgments

The authors wish to gratefully acknowledge the support provided by JPL Mobility and Robotic Systems Section for several key algorithms employed in the PALACE project. The authors also thank the US Army Aeroflight-dynamics Directorate and NASA/Army Autonomous Rotorcraft Project for their collaboration and support of the PALACE project.

References

- [1] W. Forstner. Reliability analysis of parameter estimation in linear models with application to mensuration problems in computer vision. *Computer Vision, Graphics, and Image Processing*, 40(3):273–310, 1987.
- [2] P. J. Garcia-Pardo, G. S. Sukhatme, and J. F. Montgomery. Toward vision-based safe landing for an autonomous helicopter. In *Robotics and Autonomous Systems*, volume 38, pages 19–29, 2001.
- [3] J. Ha, C. Alvino, G. Prior, M. Niethammer, E. N. Johnson, and A. R. Tannenbaum. Active contours and optical flow for automatic tracking of flying vehicles. In *IEEE ACC: Proceedings of the American Control Conference*, volume 4, pages 3441–3446. IEEE Conference Proceedings, June 2004.
- [4] J. Hintze, M. Tischler, D. Christian, T. McLain, C. Theodore, and J. Montgomery. Simulated autonomous landing of a rotorcraft unmanned aerial vehicle in a non-cooperative environment. In *Annual Forum of the American Helicopter Society*, June 2004.
- [5] A. Johnson and L. Matthies. Precise image-based motion estimation for autonomous small body exploration. In *5th International Symposium on Artificial Intelligence, Robotics, and Automation in Space (i-SAIRAS)*, Noordwijk, Netherlands, June 1999.
- [6] J. P. Lewis. Fast normalized cross-correlation. In *Vision Interface*, Canada, 1995.
- [7] H. Moravec and D. Gennery. Cart project progress report. Technical report, Stanford University, September 1976.
- [8] A. L. Peterson. Launched to return. *Unmanned Vehicles Magazine*, 1(1), January 2003.
- [9] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme. Visually guided landing of an unmanned aerial vehicle. *IEEE TRA: Transactions on Robotics and Automation*, 19(3):371–380, June 2003.
- [10] S. Saripalli and G. S. Sukhatme. Landing on a moving target using an autonomous helicopter. In *Proceedings of the International Conference on Field and Service Robotics*. IEEE, July 2003.
- [11] O. Shakernia, R. Vidal, C. S. Sharp, Y. Ma, and S. Sastry. Multiple view motion estimation and control for landing an unmanned aerial vehicle. In *IEEE ICRA: Proceedings of IEEE International Conference on Robotics and Automation*, volume 3, pages 2793–2798. IEEE, IEEE Conference Proceedings, May 2002.
- [12] C. Theodore, W. Dai, T. McLain, and M. Takahashi. Full mission simulation of a rotorcraft unmanned aerial vehicle for landing in a non-cooperative environment. In *Annual Forum of the American Helicopter Society*, June 2005.
- [13] C. Theodore, D. Rowley, A. Ansar, L. Matthies, S. Goldberg, D. Hubbard, and M. Whalley. Flight trials of a rotorcraft unmanned aerial vehicle landing autonomously at unprepared sites. In *Annual Forum of the American Helicopter Society*, June 2006.
- [14] M. Whalley, M. Freed, M. Takahashi, D. Christian, A. Patterson-Hine, G. Schulein, and R. Harris. The NASA/Army autonomous rotorcraft project. In *Annual Forum of the American Helicopter Society*, May 2003.