



All Faculty Publications

2009-4

Payload Directed Flight of Miniature Air Vehicles

Randal W. Beard

Brigham Young University - Provo, beard@byu.edu

Clark Taylor

Brigham Young University - Provo

Jeff Saunders

Brigham Young University - Provo

Ryan Holt

MIT

Timothy W. McLain

Brigham Young University - Provo, mclain@byu.edu

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



Part of the [Electrical and Computer Engineering Commons](#), and the [Mechanical Engineering Commons](#)

Original Publication Citation

Recommended Citation

Beard, Randal W.; Taylor, Clark; Saunders, Jeff; Holt, Ryan; and McLain, Timothy W., "Payload Directed Flight of Miniature Air Vehicles" (2009). *All Faculty Publications*. Paper 1504.

<http://scholarsarchive.byu.edu/facpub/1504>

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

Payload Directed Flight of Miniature Air Vehicles

Randal W. Beard^{*}, Clark Taylor[†], Jeff Saunders[‡], Ryan Holt[§], Timothy W. McLain[¶]

Brigham Young University, Provo, Utah, 84602 USA

This paper describes three applications of payload directed flight using miniature air vehicles: vision-based road following, vision-based target tracking, and vision-based mapping. A general overview of each application is given, followed by simulation and flight-test results. Results demonstrate the viability of utilizing electro-optical video imagery to directly control the air vehicle flight path to enhance performance relative to the sensing objective.

I. Introduction

Traditional air traffic control is based on external sensors detecting the position of aircraft or communicating the GPS location of an aircraft to a centralized location where routes are deconflicted and replanned. However, as the number of aircraft has proliferated and with increasing need to insert UAVs into the national airspace, there is an increased need for individual aircraft to autonomously deconflict and replan their own trajectories. Doing so will require on-board sensors and associated algorithms for payload directed flight.

For miniature air vehicles (MAVs), missions are significantly different than for larger aircraft. Due to weight constraints, MAVs can only carry a limited payload and the duration of their flight is severely constrained. In addition, MAVs typically fly close to the ground and therefore must avoid terrain such as large buildings and trees. In contrast to large aircraft, whose primary mission is usually to transport its payload, the mission objective for MAVs is almost always directly tied to its sensor platform, which is the primary payload. For example, an on-board camera system may be used to identify and track ground-based targets. As another example, MAVs are envisioned to play a key roll in supporting and protecting convoys of the future. While a MAV can easily be launched from a vehicle in the convoy, recovering the MAV at a nearby landing spot may pose a mortal threat to troops. As an alternative, an on-board camera may be used to precisely guide a MAV onto the back of a vehicle in the convoy.

It is clear that payload directed flight is critically important to the future success of unmanned air vehicles both large and small. The objective of this paper is to give an overview of research on payload directed flight performed at the BYU Magicc Lab over the past several years. While our efforts have focused almost exclusively on miniature air vehicles where computational resources are limited, the algorithms have immediate application to larger UAVs. We will overview research results in the following three areas: vision-based road following (Section III), vision-based target tracking (Section IV), and vision-based mapping (Section V).

II. MAV Airframe and Platform

This section describes the experimental hardware and software used for payload directed flight at BYU. The hardware consists of a UAV equipped with an autopilot and various sensors. The software includes a simulation program that is used to test the autopilot code, and a program that is used for interfacing with both the simulator and the actual autopilot. Figure 1 is a flowchart describing the architecture of the system.

^{*}Professor, Electrical and Computer Engineering, beard@byu.edu, AIAA Senior Member.

[†]Assistant Professor, Electrical and Computer Engineering, taylor@ee.byu.edu, AIAA Member.

[‡]Graduate Assistant, BYU Magicc Lab, saunders.jeff@gmail.com.

[§]Technical Staff, MIT Lincoln Laborator, rholt@ll.mit.edu.

[¶]Professor, Mechanical Engineering, mclain@byu.edu, AIAA Associate Fellow.

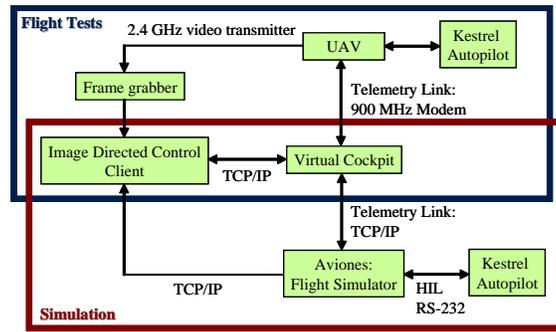


Figure 1. A flowchart depicting the layout of the basic hardware and software components used both in simulation and actual flight-tests.

A. Hardware

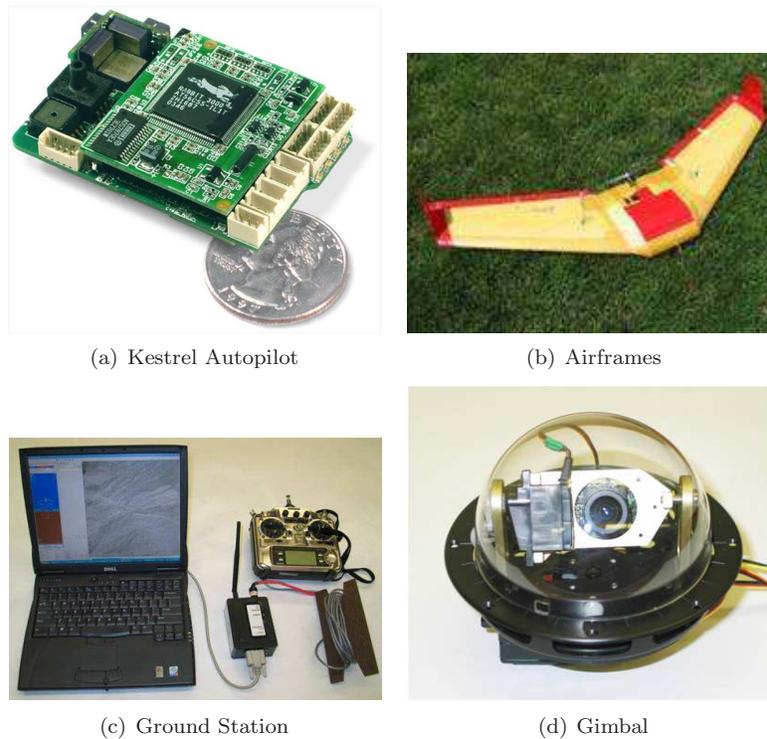


Figure 2. The hardware components used in flight-testing the algorithms described in this paper. In (a), the Kestrel Autopilot is shown, in (b), the airframe is displayed, in (c), the components of the ground station are laid out, and (d) shows the gimbal.

Figure 2 displays the main hardware components of the experimental testbed. Figure 2(a) shows the Kestrel autopilot designed at Brigham Young University (BYU) and manufactured by Procerus Technologies. It is equipped with a Rabbit 3000 29-MHz microprocessor, rate gyroscopes, accelerometers, absolute and differential pressure sensors. The autopilot measures $2.0 \times 1.4 \times 0.5$ inches and weighs 16.7 grams, making it ideal for small aircraft. Figure 2(b) shows a typical airframe used at BYU. It is a custom designed flying wing constructed of expanded polypropylene (EPP) foam and coated with Kevlar, with a wingspan of 60 inches. This airframe is easily hand-deployable and resilient to crashes, making it a good research and test vehicle. Embedded within the foam are the autopilot, batteries, a 1000 mW 900 MHz radio modem, a GPS receiver, a video transmitter, and a small analog camera mounted on a pan-tilt gimbal. The fully loaded weight is just under four pounds. With this setup the UAV can fly for approximately 100 minutes at a cruise speed of 15 m/s. The gimbal, which was designed at BYU is shown in Figure 2(d). The azimuth axis of the

gimbal has 340 degrees of range, while the elevation axis has 120 degrees of range. The camera is a Panasonic KX-141 camera with 480 lines of resolution. Figure 2(c) shows the components that comprise the ground station. A laptop runs the Virtual Cockpit ground-control software, allowing the user to interface with the UAV via a communication box. A remote control (RC) transmitter also is attached to the communication box, acting as a standby fail-safe mechanism to facilitate safe operation.

B. Software

There are two main software applications that are used in simulating and testing the developed algorithms. The first, Aviones, is a flight simulator developed at BYU, which emulates both the physics of the airframe as well as the communication between the ground station and the UAV. The motion of each virtual UAV is calculated from the full nonlinear, six-degree-of-freedom equations of motion [1]. Aviones is adaptable to many styles of aircraft and various terrain conditions. The most powerful aspect of Aviones is that the autopilot code tested in simulation is identical to the code on the actual autopilot, allowing for quick transfer from software development to flight-testing. Figure 3(a) shows a screen capture of the Aviones software during a simulated flight-test.

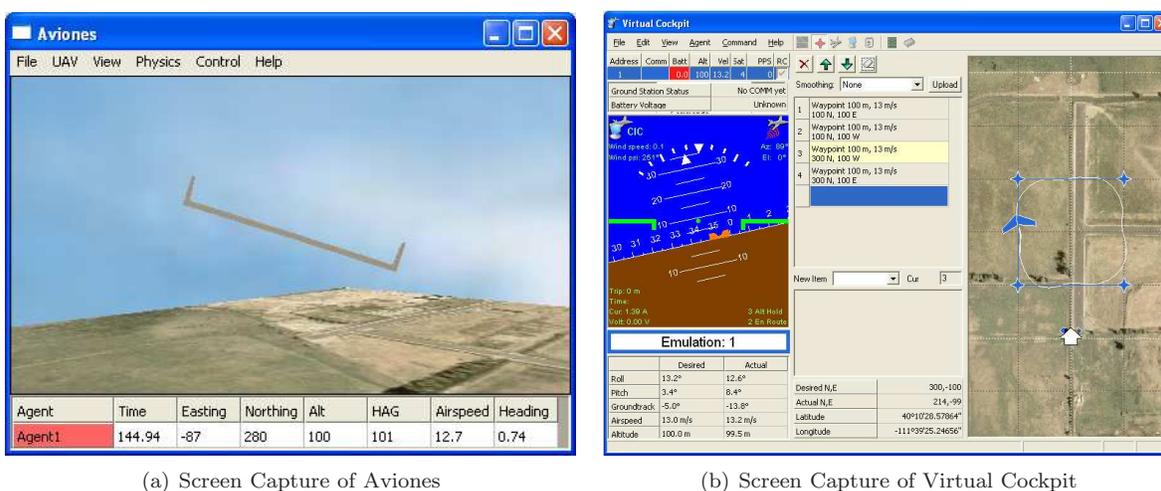


Figure 3. (a) Screen capture of Aviones with a simulated aircraft in flight. (b) Screen capture of Virtual Cockpit with the navigation screen displayed on the right-hand side. The UAV tracks the path defined by the four blue stars. The artificial horizon along with the desired and actual attitude values are located on the left-hand side.

The second piece of software that is utilized in both simulation and flight-testing is Virtual Cockpit. Virtual Cockpit connects to Aviones through a TCP/IP connection and allows the user to interface with the simulated UAV in the same manner as during a real-world flight-test. In both simulation and testing, Virtual Cockpit allows the user to view the current attitude of the UAV, tune the control loops in real-time, adjust various algorithmic variables, and command a list of waypoints for the UAV to follow. Additionally, a frame processor may be connected to Virtual Cockpit through a TCP/IP connection, allowing for video images from the UAV to be converted into control commands and sent to the UAV. Figure 3(b) shows a screen capture of the Virtual Cockpit software.

III. Vision-based Road Following

This section describes a vision-based approach to following roads and visually distinct borders. There are a variety of applications for technology including border patrol, pipeline surveillance, powerline maintenance, military convoy support, and fire monitoring [2]. Vision-based road following has been addressed in other publications [3, 4], where it has been noted that road following consists of two challenges: (1) detecting and estimating the position of the road in the image, and (2) devising a guidance strategy to follow the road.

Our approach to road detection is similar to [3] in that we use color segmentation and a connected component analysis to distinguish between road and non-road pixels. However, rather than searching the

entire image and identifying the road in the image, we simply identify the position on the boundary of the camera plane where the road flows into the image.

Our approach a guidance strategy is based on the Proportional Navigation, and Augmented Proportional Navigation schemes that are well known in the missile guidance literature [5–7]. The application of proportional navigation to road following is not immediately obvious. In contrast to missile guidance problems, our objective is not to collide with the road, but rather to track the road at a fixed altitude. Our approach is to treat the position that the road flows into the image as the target, and pursue this target as if it were at the same altitude as the UAV. A key insight is that when the road bends and curves, the apparent motion in the image plane is identical to that of an accelerating target. Therefore augmented proportional navigation can be used to track roads with significant curvature. The main features of the algorithm are described in [8].

A. Flight Test Results

This section describes some of the flight test results using the algorithm. The UAV platform used was the fixed-wing design described earlier, equipped with a pan-and-tilt gimbaled camera. Figure 4 shows the results of the image-processing algorithm, where (a) is the original video frame and (b) is the classified image. Each edge section has been processed in order to illustrate the sections sizes. Note how the middle section has not been classified. The position of the road is indicated by the red square at the top of the image.

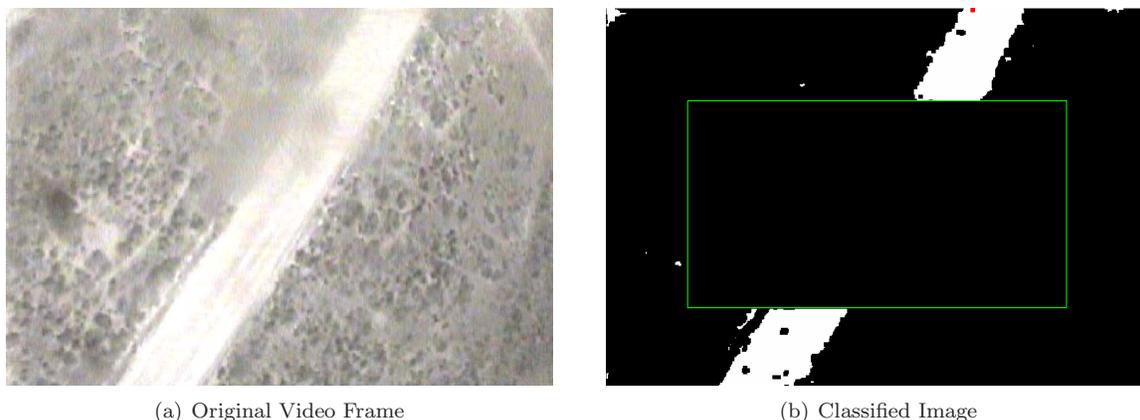


Figure 4. A sample from the results of the image-processing algorithm used during the experimental test flights. The image in (a) is the original image, while the image in (b) is the classified image. The road position determined by the algorithm for this image is indicated by the red square at the top of the image. The green lines indicate the boundaries of the sections that were processed.

In the experiment, we initialized the algorithm when the UAV was not directly over the road in order to demonstrate how the control law will cause the UAV to change its heading. Figure 5 shows an aerial view of the road that was used for the experiment, where the actual GPS location of the UAV is overlaid on top. The UAV started from the red circle and continued in a south-easterly direction over the road until reaching the red triangle. As can be seen, the control did very well.

The image tracking error is displayed in Figure 6. Note that the error converges to zero only for a short time. This deviation can be explained by two disturbances: the lag in the movement of the gimbal and high wind conditions. The lag allows the camera to leave the downward position, causing the road in the image to leave the desired location. During experiments the wind speeds were approximately 50 to 60 percent of the vehicle’s airspeed, requiring the UAV to compensate by significant rolling motions to counteract the wind. Despite these disturbances, the UAV was still able to track the road.

IV. Vision-based Target Tracking

This section briefly describes some of our results on autonomously maintaining a target in the field of view of an onboard camera with sufficient resolution for visual detection. Target tracking has a wide range of potential military and civilian uses including tracking enemy military movements, inspecting potential IED sites, and tracking criminal suspects.

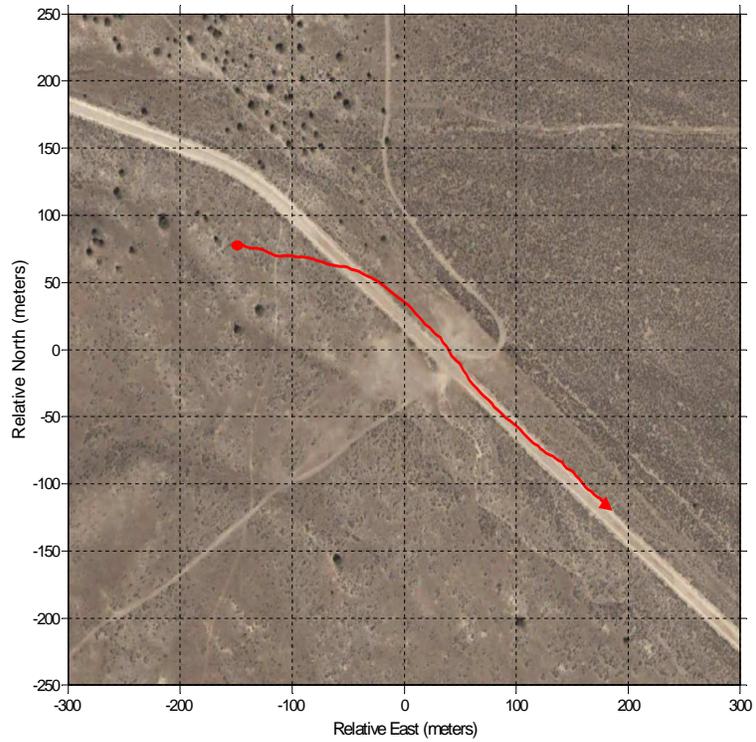


Figure 5. An aerial view of the road used in the flight test with the actual GPS path of the UAV depicted in red, starting at the circle and ending at the triangle. The image was obtained from Google Earth.

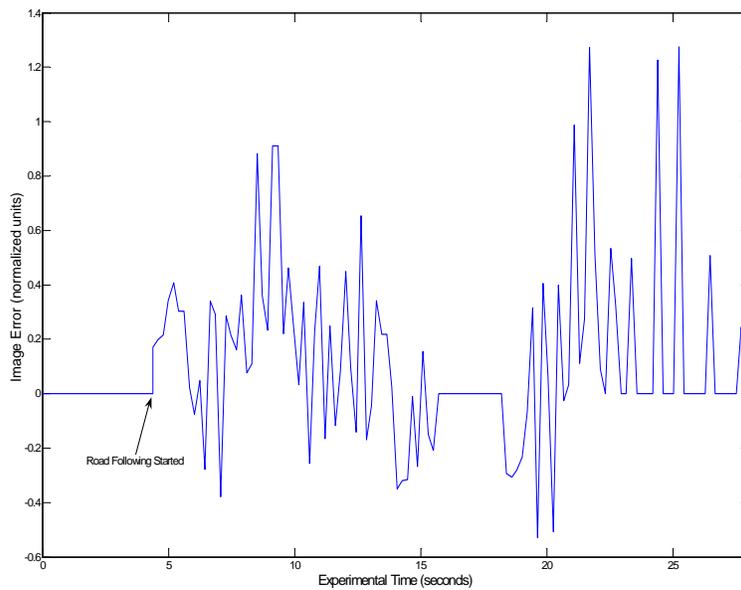


Figure 6. Image tracking error recorded during the road-following flight-test.

Target tracking is technically difficult for several reasons. Processing visual data often requires computational resources that are not available on small MAVs, thus requiring a nearby ground station to acquire and process video. Transmitted video is often noisy and intermittent because of transmission interference resulting in dropped frames and thus noise in the tracked position of the target in the image frame. This is compounded by camera jitter caused by the motion of the MAV due to air turbulence. Therefore the tracking method must be robust to target position error introduced by the camera. Camera motion caused by gusts of wind, or target movement, can cause the target to move outside the camera field of view. This implies the tracking method must also have quick response to target movement in the image plane. The overriding objective is to maintain the target in the camera field of view indefinitely.

Previous work has addressed some of the problems described in the previous paragraph. For example, Thomasson derived a solution to target tracking in wind [9] by assuming an elevation controlled camera and constant wind. He found that a MAV must fly an ellipse with the target located at one of its foci and with the major axis of the ellipse aligned perpendicular to the direction of the wind. The calculation of the dimensions of the ellipse is based on prior knowledge of the magnitude and direction of the wind velocity. Given the kinematic constraints of the MAV, the wind velocity, and the target velocity, an elliptical trajectory is generated for the MAV to track. As the wind velocity and target velocity remain constant, the MAV may track the target indefinitely. Feedback is not incorporated in the solution, resulting in open-loop target tracking. Problems arise as wind velocity changes, gusts occur, or the target changes course, causing the target to leave the camera’s field of view. This paper improves on [9] by incorporating a feedback method where the resulting path in wind is an elliptical orbit around the target.

Successful path planning feedback solutions to target tracking problems have been demonstrated in [10–17]. The general approach is to generate paths based on current knowledge. As information is acquired, whether about the wind, target, or MAV, the path to track the target is regenerated. Reference [15] generates paths to a landing position based on vision of a runway or other landmark. Moving targets are tracked using dynamic planning and gimballed cameras in [10–14, 16]. Unfortunately, gimbals are not always feasible on small MAVs. Removing the gimbal introduces additional kinematic constraints and path planning becomes more difficult. Additionally, a gust of wind or a sudden change in course of the target may push the target outside the field of view before a response is generated, possibly causing the MAV to lose the target altogether. Fast response is necessary to prevent the target from leaving the image plane. Our approach differs from previous work by utilizing target movement in the image plane to maneuver the MAV such that the target remains in the image plane. Reaction to movement in the image plane generates a response that prevents the target from leaving the image plane.

Our basic approach, which is described more fully in [18], is to use a nonlinear control law that uses range-to-target and bearing-to-target information obtained from target motion in the image plane to maintain the target in the image plane. As the target moves in the image plane, the control law “pushes” the target back to the center of the camera field of view.

A. Simulation Results

Simulations were conducted in Aviones using a six degree-of-freedom model. We used an emulated Kestrel autopilot [19] with airspeed set to $V_a = 13$ m/s. The location and size of the target in the camera image plane were calculated and delayed to emulate vision processing. The results for a MAV flying clockwise around stationary target in zero wind are shown in Figure 7 (a). Without wind, the expected shape of the orbit is circular.

The results for clockwise flight around a stationary target in a constant wind of $V_w = 5$ m/s to the North are shown in Figure 8 (a). Notice that the flight path converges to an ellipse as was predicted in [9], without explicit path planning. The corresponding motion of the target in the image plane is shown in Figure 8 (b).

B. Flight Results

Flight tests were conducted using the testbed described in Section II. The camera was mounted at an azimuth angle of 90 degrees (out the right wing) and an elevation angle of 30 degrees. The wind speed during the flight tests was approximately 2 m/s. The video was transmitted to the ground station via a 2.4 GHz analog transmitter, where a color segmentation algorithm was used to locate the target in the image frame. The target was a red cloth on the ground. The color segmentation algorithm used thresholds to remove all colors

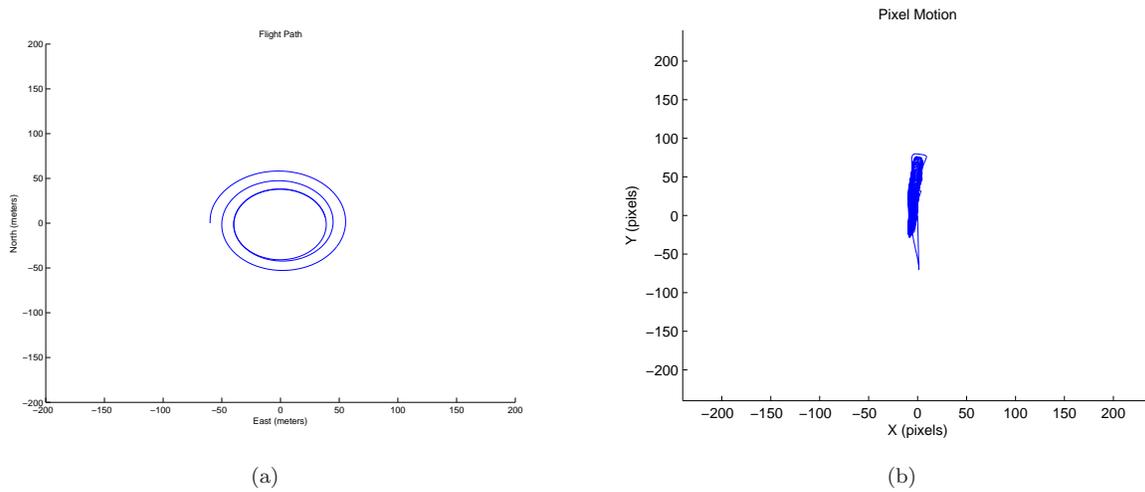


Figure 7. (a) Simulated flight path in zero wind. The MAV's initial configuration allow the camera to capture the target. The MAV navigates onto a stable clockwise orbit around the target. (b) The corresponding motion of the target in the image plane.

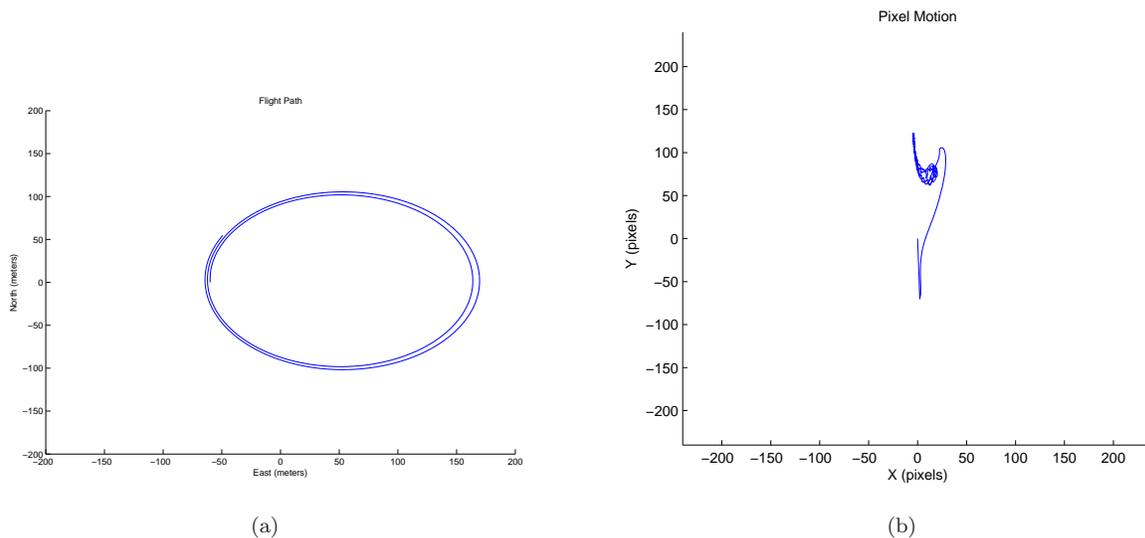


Figure 8. (a) Simulated flight path result with wind of 5m/s to the North. (b) The corresponding motion of the target in the image plane.

other than red, and returned the location of the largest patch of red. A representative image from the video sequence is shown in Figure 9.



Figure 9. The ground station processes video by color segmentation to find the location of the red cloth.

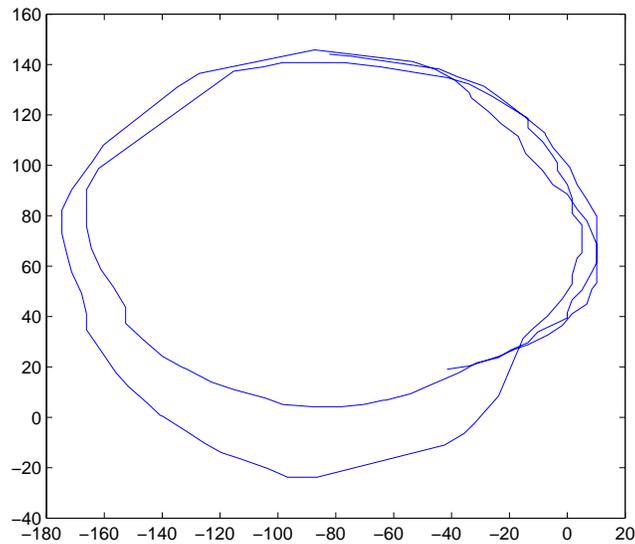


Figure 10. Flight path in constant ambient wind of approximately $V_w = 2$ m/s.

The resulting flight path of the MAV is shown in Figure 10. The ambient wind resulted in an elliptical orbit with eccentricity $\epsilon = 0.17$. The theoretical prediction for a wind of $V_w = 2$ m/s is $\epsilon = \frac{V_w}{V_a} = 0.15$. The guidance algorithm presented in this paper successfully maintained the target in the field-of-view of the camera throughout the flight test.

V. Vision-based Mapping

This section briefly describes our work in mapping an area observed by an MAV. To create a map of an area observed by a MAV, two technical questions must be addressed: (1) how to create a composite image

from the individual video frames collected by the MAV and (2) how to precisely geo-locate the composite image. The ability to create a composite image from several video frames is usually referred to as mosaicking and has been extensively addressed in prior work. A good overview of mosaicking can be found in [20].

Once a mosaic has been created, the next step in mapping is to correctly geo-register the mosaic. Prior work has followed one of two approaches to registering the mosaic to its geo-coordinates. First, after the mosaic has been created, it is “registered” with prior geo-referenced imagery to determine its geo-location [21, 22]. The problems with this approach are: (1) this requires that mosaicking be completed before geo-registration can occur, leading to significant delays in when the geo-referenced mosaic becomes available, (2) pre-existing geo-referenced imagery may not be available in sufficient detail to enable alignment with MAV video, and (3) there are often significant differences in appearance between pre-existing and MAV imagery due to environmental factors which inhibits registration. For example, if the pre-existing imagery was captured during the winter when snow was on the ground while the MAV video is captured in the summer, there will be significant differences between the two sets of imagery making registration difficult.

The second approach used to geo-register mosaics is to take telemetry information generated by a GPS/INS unit on-board the MAV to estimate the location of the camera for one of the captured video frames. From this information, the projection of that image onto the ground can be computed, enabling geo-registration of the entire mosaic. The difficulty with this approach is that due to the low-quality sensors on-board a MAV, each individual pose estimate can have a significant amount of error, leading to significant deviations in the location of an image on the ground. For example, in Figure 11, we show several images that were projected onto the ground plane using only the most recent pose estimate of the MAV. As shown in Figure 11, the placement of the images varies significantly with noise in the pose estimates, causing serious discontinuities between video frames.

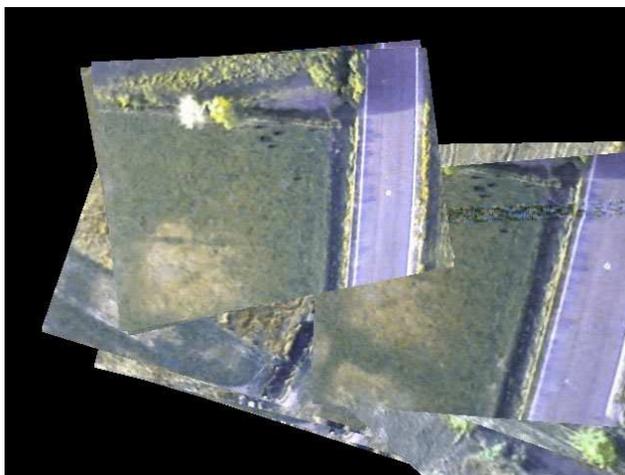


Figure 11. A figure demonstrating the effect of noisy pose estimates on providing context to video frames. Due to noise in the pose estimates, ground location estimates vary significantly from frame to frame, causing, among other effects, significant discontinuities between images.

Our basic approach to mapping (described more fully in [23]) is to filter together estimates of the mosaic’s geo-location together over time. Each individual estimate is generated using the GPS/INS system (the Kestrel autopilot shown in Figure 2(a)) and an Unscented Kalman Filter (UKF) is used to enable filtering of these estimates over time. Because a UKF is used, the best mosaic so far is always available to the user of the system, overcoming the time delays associated with associated with registering to pre-existing imagery. Because multiple estimates of geo-location are filtered together, the overall accuracy of the geo-location estimate is greatly improved over traditional methods based on GPS/INS systems, enabling accurate placement of the mosaic in a geo-coordinate system.

A. Mapping Results

To demonstrate the effectiveness of our method, we first created a mosaic assuming that the first pose estimate received from the MAV was perfectly correct. The mosaic created consists of over 150 frames of data spanning an area over 250 meters long. A picture of this mosaic, overlaid on Google Earth to show



(a) Mosaic created assuming accurate first pose estimate

(b) Mosaic created with novel geo-location filter

Figure 12. Two pictures demonstrating the geo-registration accuracy of our technique. Mosaics are placed against a Google Earth background for the same area to demonstrate geo-referencing accuracy. In subfigure (a), only the initial telemetry is used to geo-reference the mosaic. In subfigure (b), our geo-location filter is applied, leading to a decrease in geo-location errors of 75 to 5 meters.

geo-location accuracy, is shown in Figure 12(a). Note that due to errors in the initial pose estimate (the MAV was flying from North to South), the mosaic is very poorly geo-referenced towards the bottom of the mosaic. Geo-location error of the circled house is approximately 75 meters.

The result of creating a mosaic from the same set of data utilizing our proposed geo-location filter is shown in Fig. 12(b). Note that this mosaic is (a) geo-registered much more accurately than the mosaic in Fig. 12(a) (e.g., the house geo-location now has 5 meters of error), (b) contains much less perspective distortion than the image in Fig. 12(a), and (c) does not contain significant discontinuities in the mosaic, demonstrating the ability to mosaic and add context to the mosaic simultaneously.

VI. Conclusions

In this paper, we have discussed three different applications where the flight path of a MAV was controlled by the imagery obtained from an electro-optical video camera payload. The flight-test results obtained demonstrate the feasibility of using a low-cost sensor payload to direct the flight of a MAV.

VII. Acknowledgments

This research was supported by NASA under STTR contract No. NNA04AA19C to Scientific Systems Company, Inc. and Brigham Young University, the Air Force Research Laboratory Munitions Directorate under SBIR contract No. FA8651-07-C-0094 to Scientific Systems Company, Inc. and Brigham Young University, and AFOSR Young Investigator Award No. FA9550-07-1-0167.

References

- ¹Stevens, B. L. and Lewis, F. L., *Aircraft Control and Simulation*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2nd ed., 2003.
- ²Casbeer, D. W., Kingston, D. B., Beard, R. W., McLain, T. W., Li, S.-M., and Mehra, R., "Cooperative Forest Fire Surveillance Using a Team of Small Unmanned Air Vehicles," *International Journal of System Sciences*, Vol. 37, No. 6, May 2006, pp. 351–360.
- ³Frew, E., McGee, T., Kim, Z., Xiao, X., Jackson, S., Morimoto, M., Rathinam, S., Padiyal, J., and Sengupta, R., "Vision-Based Road-Following Using a Small Autonomous Aircraft," *Proceedings of the IEEE Aerospace Conference*, Vol. 5, March 2004, pp. 3006–3015.
- ⁴Rathinam, S., Kim, Z., Soghikian, A., and Sengupta, R., "Vision Based Following of Locally Linear Structure using an Unmanned Aerial Vehicle," *Proceedings of the 44th IEEE Conference on Decision and Control and the European Control*

Conference, Seville, Spain, December 2005, pp. 6085–6090.

⁵Zarchan, P., *Tactical and Strategic Missile Guidance*, Vol. 199 of Progress in Astronautics and Aeronautics, American Institute of Aeronautics and Astronautics, Inc., Reston, Virginia, 4th ed., 2002.

⁶Lin, C. F., *Modern Navigation, Guidance, and Control Processing*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.

⁷Siouris, G. M., *Missile Guidance and Control Systems*, Springer-Verlag, New York, 2004.

⁸Holt, R. S. and Beard, R. W., “Vision-Based Road-Following Using Proportional Navigation,” *IEEE Transactions on Control Systems Technology*, (in review).

⁹Thomasson, P. G., “Guidance of a Roll-Only Camera for Ground Observation in Wind,” *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 1, January–February 1998, pp. 39–44.

¹⁰Rezza, R., “Path Following for Air Vehicles in Coordinated Flight,” *International Conference on Advanced Intelligent Mechatronics*, September 1999.

¹¹Lee, J., Huang, R., Vaughn, A., Xiao, X., and Hedrick, J. K., “Strategies of Path-Planning for a UAV to Track a Ground Vehicle,” *Proceedings of the 2nd Annual Autonomous Intelligent Networks and Systems Conference*, 2003.

¹²Stolle, S. and Rysdyk, R., “Flight path following guidance for unmanned air vehicles with pan-tilt camera for target observation,” *Digital Avionics Systems Conference*, 2003.

¹³Rysdyk, R., “Unmanned Aerial Vehicle Path Following for Target Observation in Wind,” *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 5, September–October 2006, pp. 1092–1100.

¹⁴Frew, E., Xiao, X., Spry, S., McGee, T., Kim, Z., Tisdale, J., Sengupta, R., and Hendrick, J. K., “Flight Demonstrations of Self-Directed Collaborative Navigation of Small Unmanned Aircraft,” *AIAA 3rd Unmanned Unlimited Technical Conference*, September 2004.

¹⁵Frezza, R. and Altafini, C., “Autonomous landing by computer vision: an application of path following in SE(3),” *Conference on Decision and Control*, December 2004.

¹⁶Stepanyan, V. and Hovakimyan, N., “Visual Tracking of a Maneuvering Target,” *AIAA Guidance, Navigation and Control Conference and Exhibit*, August 2006.

¹⁷Sinopoli, B., Micheli, M., Donatot, G., and Koo, T. J., “Vision Based Navigation for an Unmanned Aerial Vehicle,” *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*, 2001, pp. 1757–1764.

¹⁸Saunders, J. and Beard, R., “Tracking a Target in Wind Using a Micro Air Vehicle with a Fixed Angle Camera,” *Proceedings of the American Control Conference*, Seattle, WA, June 2008, pp. 3863–3868.

¹⁹“Procerus Technologies,” <http://procerusuav.com/>.

²⁰Szeliski, R. and Shum, H.-Y., “Creating Full View Panoramic Image Mosaics and Environment Maps,” *SIGGRAPH*, 1997, pp. 251–258.

²¹Kumar, R., Sawhney, H., Asmuth, J. C., Pope, A., and Hsu, S., “Registration of video to geo-referenced imagery,” *Fourteenth International Conference on Pattern Recognition*, Vol. 2, Aug 1998, pp. 1393–1400.

²²Kumar, R., Samarasekera, S., Hsu, S., and Hanna, K., “Registration of highly-oblique and zoomed in aerial video to reference imagery,” *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, Vol. 4, Sep 2000, pp. 303–307.

²³Taylor, C. N. and Andersen, E. D., “An Automatic System for Creating Geo-referenced Mosaics from MAV Video,” *IEEE International Conference on Intelligent Robots and Systems*, 2008.