



AFRL-RW-EG-TR-2012-091

Egomotion Estimation with Optic Flow and Air Velocity Sensors

Adam J Rutkowski
Mikel M Miller
Roger D Quinn
Mark A Willis

Air Force Research Laboratory
Munitions Directorate
101 W Eglin Blvd.
Eglin AFB, FL 32542

17 September 2012

INTERIM REPORT

DISTRIBUTION A Approved for public release; distribution unlimited. (96ABW-2011-0322)

**AIR FORCE RESEARCH LABORATORY
MUNITIONS DIRECTORATE**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

Qualified requestors may obtain copies of this report from the Defense Technical Information Center (DTIC) <<http://www.dtic.mil/dtic/index.html>>.

.

AFRL-RW-EG-TR-2012-091 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

NICHOLAS I. RUMMELT
Acting Technical Advisor, RWW

ADAM J. RUTKOWSKI
Program Manager

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE				<i>Form Approved OMB No. 0704-0188</i>	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY)		2. REPORT TYPE		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (Include area code)

List of Tables

1	Methods of computing apparent egovelocity from apparent egodisplacement	8
2	Formulation of the vector of $\hat{\xi}$, $\hat{\mathbf{V}}_g$, $\mathbf{L}$	10
3	Methods of computing the ground vector, $\mathbf{v}_g$, from position data, $\mathbf{p}$	12

List of Figures

1	Problem Scenario	5
2	Processing model block diagram	5
3	Flight track of a pheromone tracking moth	11
4	Height estimation error with zero sensor noise	14
5	Height estimation error with optic flow measurement noise	15
6	Height estimation error with air velocity measurement noise	16
7	Height estimation error over flat ground in naturalistic wind	17
8	Height estimate vs. time	18
9	Height estimation error over inclined in naturalistic wind	19
10	Height and ground slope estimates vs. time	20
11	Ground vector and wind vector estimates	21

Egomotion Estimation with Optic Flow and Air Velocity Sensors

Adam J. Rutkowski¹

Mikel M. Miller¹

Roger D. Quinn²

Mark A. Willis²

¹*Air Force Research Laboratory, Eglin AFB, FL, 32542, USA*

²*Case Western Reserve University, Cleveland, OH, 44106, USA*

Abstract

We develop a method that allows a flyer to estimate its own motion (egomotion), the wind velocity, ground slope, and flight height using only inputs from onboard optic flow and air velocity sensors. Our artificial algorithm demonstrates how it could be possible for flying insects to determine their absolute egomotion using their available sensors, namely their eyes and wind sensitive hairs and antennae. Although many behaviors can be performed by only knowing the direction of travel, behavioral experiments indicate that odor tracking insects are able to estimate the wind direction and control their absolute egomotion (i.e. groundspeed). The egomotion estimation method that we have developed, which we call the opto-aeronautic algorithm, is tested in a variety of wind and ground slope conditions using a video recorded flight of a moth tracking a pheromone plume. Over all test cases that we examined, the algorithm achieved a mean absolute error in height of 7% or less. Furthermore, our algorithm is suitable for the navigation of aerial vehicles in environments where signals from the Global Positioning System are unavailable.

1 Introduction

Flying insects navigate proficiently through their environment using multiple sensory modalities and limited computational resources (Taylor and Krapp, 2008). The use of multiple sensor modalities allows insects to navigate in a wide variety of environmental conditions. Thus, flying insects can serve as inspiration for methods and sensory systems that can be used to navigate unmanned aerial vehicles.

Flying insects use their compound eyes to detect optic flow, or the apparent motion of visual patterns in their visual field. The optic flow at any point in the visual field is a combination of the insect's own translational and rotational motion, or egomotion, and the motion, if any, of the visual pattern at that part of the visual field. Optic flow is essentially an angular rate of the relative motion of a visual pattern in a given viewing direction. There are several artificial methods of computing optic flow from a vision sensor. Barron et al. (1994) group these techniques into differential, region-based matching, energy-based, and phase-based techniques. These techniques vary in computational complexity, and the accuracy of these techniques can be dependent on the visual structure of the scene. In our work, we are not particularly concerned with how optic flow is computed. Instead, we will focus on using optic flow in higher levels of processing to estimate egomotion.

At the next level of processing, the optic flow field, which consists of a collection of optic flow vectors along multiple viewing directions, is used to estimate egomotion. If everything in the environment is stationary, the rotational egomotion and the *direction* of translational egomotion can be computed from the optic flow field, but the *magnitude* of translational egomotion (i.e. groundspeed) is inversely related to the distance of the scene from the visual system (Koenderink and van Doorn, 1987). For instance, Miao et al. (1996) implemented a differential optic flow technique to estimate these quantities from a recorded image sequence from a forward looking camera mounted aboard a helicopter. The computation of egomotion from a spherical motion field has been formulated by Koenderink and van Doorn (1987), and a simpler, noniterative procedure was developed by Dahmen et al. (2001). Srinivasan (1994) developed the image interpolation technique to estimate egomotion directly from a sequence of image intensity patterns from a monocular camera without

This paper appeared in Biological Cybernetics 104(6): 351-367

the intermediate step of computing the optic flow in multiple viewing directions. This technique was extended by Bab-Hadiashar et al. (1995) and Nagle and Srinivasan (1996) to allow for the estimation of the component of translational egomotion perpendicular to the visual system, and further extended to the full 6 degree-of-freedom case by Nagle and Srinivasan (1997). However, the fundamental problem with using vision to estimate egomotion is that the groundspeed cannot be determined unless the distance to features in the visual field is known or can be measured or estimated.

In the UAV literature, several approaches have been taken to resolve the scale ambiguity problem that is inherent with a visual system. Recently, Çelik et al. (2009) have proposed an approach for absolute distance and groundspeed estimation using only a monocular camera; however, it is assumed that the flight height is known. Franz et al. (2004) have developed a method of distance and groundspeed estimation using an omnidirectional camera, but knowledge of the average scene distance is required. Flight height can be measured with a downward looking distance measurement sensor, such as a sonar sensor (Conroy et al., 2009) or a laser rangefinder (Garratt and Chahl, 2008). Another approach is to use stereo vision (Toupet et al., 2007). It has also been common to use an inertial measurement unit to aid a vision based navigation system to determine the distance to the scene. However, none of these engineering solutions is used by flying insects. Since insects have fixed focus optics, closely spaced eyes, and typically only a small region of visual overlap between their two eyes, their stereoscopic vision is only useful within a range of a couple of centimeters (Srinivasan et al., 1999). Insects possess sensors for measuring angular rates (Pringle, 1948; Sane et al., 2007); however, a sensor for measuring linear accelerations has not yet been found in insects. If a flying insect could somehow estimate range by combining the information from its visual system with information from another of its available sensors, then it could also estimate the magnitude of its translational egomotion.

Flying insects use a combination of their antennae and various types of mechanosensory hairs to detect air currents (Gewecke, 1974). As an insect flies, the air current that it senses is a combination of its egomotion and the wind. In the biological literature, this relationship is referred to as the triangle of velocities (Marsh et al., 1978). In aeronautics, it is simply called the wind triangle (Comeaux, 1983). The velocity of the insect relative to a stationary point on the ground is the ground vector, $\mathbf{v}_g$, the velocity of the air mass relative to a stationary global reference frame is the wind vector, $\mathbf{w}$, and the velocity of the insect relative to the air mass is the air vector, $\mathbf{v}_a$. The wind vector, ground vector, and air vector are related at any point in time, t , through the vector equation (1).

$$\mathbf{w}(t) = \mathbf{v}_g(t) - \mathbf{v}_a(t) \quad (1)$$

There is evidence that some insects can control their groundspeed in an absolute sense, particularly in insects that use pheromones to attract and find mates. In several insect species, a male insect locates a mate by flying upwind when it smells the sex-attractant pheromone released by a female of the same species (Kennedy and Marsh, 1974). The speed at which they progress upwind while tracking an odor (i.e. the upwind component of the ground vector) is nearly the same in any wind speed, thus these insects do not fly at a preferred airspeed (Willis and Arbas, 1991; Zanen and Cardé, 1996; Foster and Howard, 1999). If odor tracking insects were flying so as to maintain a preferred optic flow across their retinas, then it would be expected that the ground vector would increase proportionally with odor plume altitude. However, a proportional relationship between the ground vector and odor plume altitude has never been observed for any odor tracking insect [moths *Heliothis virescens* and *Grapholita molesta* (Kuenen and Baker, 1982), *Epiphyas postvittana* (Foster and Howard, 1999), beetles *Prostephanus truncatus* (Fadamiro et al., 1998), or wasps *Microplitis croceipes* (Zanen and Cardé, 1996)]. Furthermore, during odor tracking, an insect must be able to resolve the wind vector from the air vector, thus, from (1), it must also be able to determine its ground vector.

We have developed a method that simultaneously estimates egomotion, wind velocity, flight height, and ground slope using measurements from onboard air velocity and optic flow sensors. We call this algorithm the opto-aeronautic algorithm. The validity of our estimator is tested in a variety of wind and ground slope conditions.

2 Approach

We divide our approach into three sections. First, we present a simplified formulation of the opto-aeronautic algorithm. Next, we present an extended formulation of the algorithm that relaxes several of our assumptions that were made in our simplified formulation. Finally, we present the procedure that we use to test the extended formulation of the algorithm using simulated and experimentally collected data.

2.1 Simplified formulation

To introduce our approach, we begin with the simplified problem of determining egomotion when flying directly forward in a constant wind over a flat ground. Onboard sensors measure the air vector and the optic flow field. To estimate egomotion, we make the following assumptions, which will also be true in the extended formulation.

1. The air vector and optic flow are measured simultaneously and sampled at a constant rate. This assumption is made only to simplify the development of the algorithm. The time between samples, or time step length, is Δt .
2. There is no lag or bias in the optic flow and air velocity sensors. Any real sensor will have lag, however, any lag that is present in the sensors will simply introduce lag in the estimation of egomotion, but will not affect the quality of the estimate.
3. All visible objects are stationary, thus any optic flow is induced solely through the motion of the visual system. If the field of view of the visual system is sufficiently wide, then objects that are moving in the environment will contribute very little to the estimation of egomotion.
4. Only the downward looking portion of the optic flow field is used to estimate egomotion since this portion of the visual field is known to contribute to speed control in flying insects (Kennedy, 1940; Kennedy and Marsh, 1974).

We define the apparent egovelocity vector, γ , as an estimate of the apparent velocity scaled with respect to the current height. The apparent egovelocity is obtained directly from the optic flow field, for example, by using the image interpolation technique of Srinivasan (1994). In our simplified problem, we assume that the ground vector, or in this case, groundspeed, is directly equal to the product of the apparent egovelocity and the height. Thus, the wind is related to the apparent egovelocity, height, and airspeed through equation (2).

$$w = \gamma h - v_a \quad (2)$$

Notice that this equation has two unknown quantities, w and h , which cannot be uniquely determined at a single instant in time. We could uniquely solve for estimates of the wind speed and flight height, $\hat{w}$ and $\hat{h}$, by using (2) to create a system of two equations at two different points in time. However, the estimates for wind speed and flight height would be highly dependent on noise in the measurements of γ and v_a . To reduce the effects of sensor measurement noise on the estimation process, we solve for $\hat{w}$ and $\hat{h}$ in a least squares sense. First, we collect measurements of γ and v_a over a short length of time and store them in batch vectors, $\mathbf{\Gamma}$ and $\mathbf{V}_a$, as given in (3) and (4). These batch vectors can be thought of as a short-term memory of the sensor measurements. We refer to this length of time as the batch window, which contains n measurements.

$$\mathbf{\Gamma} = [\gamma(t_1) \ \dots \ \gamma(t_n)]^T \quad (3)$$

$$\mathbf{V}_a = [v_a(t_1) \ \dots \ v_a(t_n)]^T \quad (4)$$

From (2), we formulate an estimate of the wind over all points in the batch window, $\hat{\mathbf{W}}$, in the following manner.

$$\hat{\mathbf{W}} = \mathbf{\Gamma} \hat{h} - \mathbf{V}_a \quad (5)$$

Since we know in this case that the wind is constant, we can use (6) as a model for the wind, where $\check{\mathbf{W}}$ is a batch vector that contains the value of the wind model at each point in the batch window, b_w is an unknown wind model coefficient, and $\mathbf{1}$ is a column vector of all ones.

$$\check{\mathbf{W}} = \mathbf{1}b_w \quad (6)$$

Ideally, the wind estimate, $\hat{\mathbf{W}}$, and the wind model, $\check{\mathbf{W}}$, should be equal at all points in the batch window. However, since there may be noise in the sensors, we solve for values of b_w and $\hat{h}$ that minimize, in a least squares sense, the difference between the wind estimate and the wind model. If we define the wind error, E_w , as in (7), where $\|\cdot\|$ represents the 2-norm or Euclidean norm, we can formulate a linear least squares problem, (8), to solve for b_w and $\hat{h}$, which for our simplified problem takes the form of (9).

$$E_w = \|\hat{\mathbf{W}} - \check{\mathbf{W}}\| \quad (7)$$

$$\min E_w^2 \quad (8)$$

$$\min_{\hat{h}, b_w} \left\| \mathbf{1}b_w - \mathbf{\Gamma}\hat{h} - \mathbf{V}_a \right\|^2 \quad (9)$$

To solve the minimization problem of (9), first assume that the height is known. Using the Moore-Penrose pseudoinverse, which is defined in (10) for any matrix $\mathbf{T}$ with more rows than columns, the wind coefficient is obtained from (11).

$$\mathbf{T}^+ = (\mathbf{T}^T \mathbf{T})^{-1} \mathbf{T}^T \quad (10)$$

$$b_w = \mathbf{1}^+ (\mathbf{\Gamma}\hat{h} - \mathbf{V}_a) = \frac{1}{n} \mathbf{1}^T (\mathbf{\Gamma}\hat{h} - \mathbf{V}_a) \quad (11)$$

By substituting (11) back into (9), and defining a matrix $\mathbf{\Theta}$ as in (12), where $\mathbf{I}$ is an $n \times n$ identity matrix, we can now solve for $\hat{h}$ according to (13).

$$\mathbf{\Theta} = \frac{1}{n} \mathbf{1}\mathbf{1}^T - \mathbf{I} \quad (12)$$

$$\hat{h} = \frac{\mathbf{\Gamma}^T \mathbf{\Theta}^T \mathbf{\Theta} \mathbf{V}_a}{\mathbf{\Gamma}^T \mathbf{\Theta}^T \mathbf{\Theta} \mathbf{\Gamma}} \quad (13)$$

A solution for the height estimate can only be obtained if the groundspeed is non-constant. Otherwise, $\mathbf{\Gamma}$ will be in the null space of $\mathbf{\Theta}$, thus the product of $\mathbf{\Theta}$ and $\mathbf{\Gamma}$ will be zero and the denominator of (13) will be zero. In effect, *changes* in apparent egoveLOCITY must be correlated with *changes* in airspeed to deduce the effect of the constant wind. The groundspeed is determined by multiplying $\hat{h}$ by γ , and the wind is determined by substituting (13) back into (2).

2.2 Extended formulation

We now extend our formulation to consider the case of motion in three translational degrees-of-freedom and one rotational degree-of-freedom about the y-axis (yaw) (Figure 1). Methods for overcoming the requirements of zero pitch rate and zero roll rate are discussed later, but not explicitly included in the formulation. We remove the requirement of a constant wind by introducing a parameterized wind model. We will also examine the effects of noise in the sensor measurements on the quality of the egomotion estimates. In addition, we will implement a more complex optic flow model that accounts for the effects of discrete time sampling. Furthermore, we will extend the basic approach to handle flight over a sloped ground.

Our opto-aeronautic algorithm is summarized in the block diagram in Figure 2. The algorithm is divided into three sections - air velocity preprocessing, optic flow preprocessing, and sensor fused estimation. The air velocity preprocessing and optic flow preprocessing are independent and may be performed simultaneously.

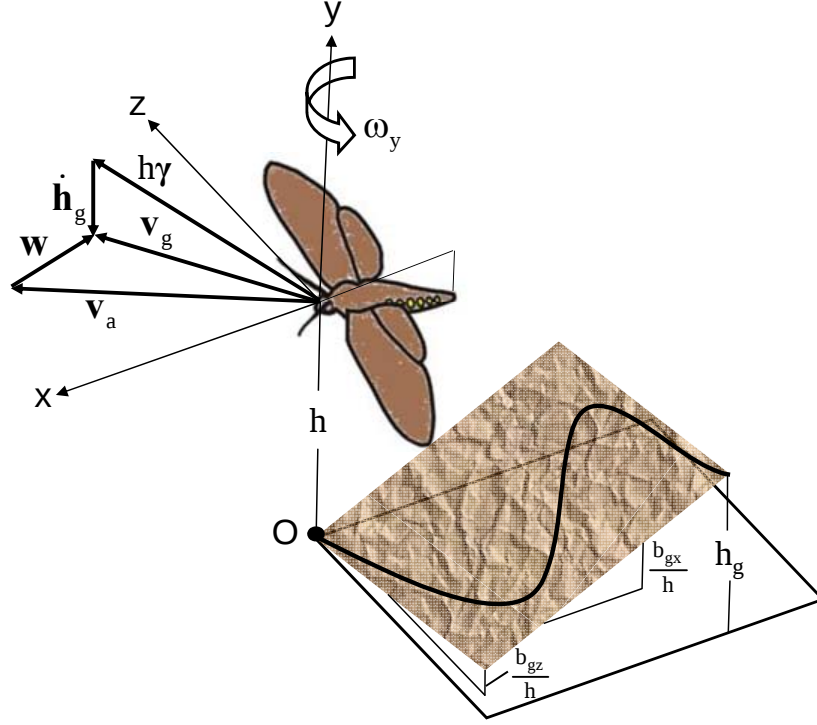


Figure 1: A flyer traverses over an inclined ground. The trajectory flown is not necessarily straight, as indicated by the curved path across the ground. All variables are described in the text.

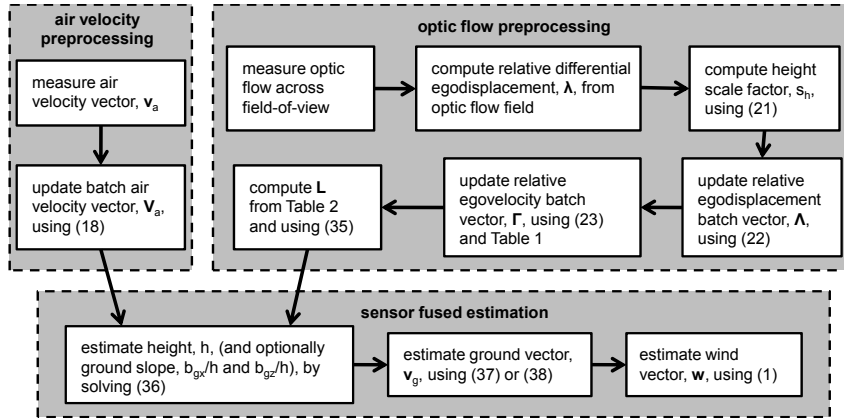


Figure 2: Block diagram of the information processing model of the opto-aeronautic algorithm

Measurements of the air vector and optic flow are made in a body-fixed reference frame. We use a right-handed Cartesian coordinate system to represent the sensor measurements. The coordinate system is fixed to the visual system such that the x-axis is aligned with a horizontal projection of the longitudinal axis of the flyer, the positive y-axis is directly upward (opposes the gravity vector), and the positive z-axis points out the right side of the body (Figure 1). The first, second, and third components of the air vector and the optic flow vector are the x-component, y-component, and z-component, respectively. We place a caret (^) above estimated quantities to distinguish them from "true" quantities.

In contrast to the simplified problem, we no longer necessarily assume that the wind speed and direction are constant. Instead, we assume that the wind is temporally smooth over a short time interval (the batch window) and the terrain is spatially smooth. To implement the smoothness constraints, we use a sliding batch, or moving horizon, estimation process in discrete time and we use batch vectors to organize the sensor measurements and state estimates. Let $\mathbf{V}$ represent a batch vector that can be replaced with $\mathbf{V}_a$ (batch air vector), $\mathbf{V}_g$ (batch ground vector), or $\mathbf{W}$ (batch wind vector). The vector $\mathbf{V}$ with subscript e_k contains the n_V most recent sensor measurements in the direction of e_k , as in (14), where the directions e_1 , e_2 , and e_3 are equivalent to the coordinate system axes x, y, and z, respectively. The batch vectors for all Cartesian axes are then stacked to form an augmented batch vector as in (15).

$$\mathbf{V}_{e_k} = \begin{bmatrix} v_{e_k}(t - (n_V - 1)\Delta t) \\ \vdots \\ v_{e_k}(t - \Delta t) \\ v_{e_k}(t) \end{bmatrix} \quad k = 1, 2, 3 \quad (14)$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{e_1} \\ \mathbf{V}_{e_2} \\ \mathbf{V}_{e_3} \end{bmatrix} = \begin{bmatrix} \mathbf{V}_x \\ \mathbf{V}_y \\ \mathbf{V}_z \end{bmatrix} \quad (15)$$

We use a left-pointing arrow ($\leftarrow$) to indicate an update of the batch vectors when proceeding from one time step to the next. For example, the following statement indicates that the j^{th} element of the vectors $\mathbf{V}_x$, $\mathbf{V}_y$, and $\mathbf{V}_z$ is determined by multiplying b by the $(j+1)^{\text{th}}$ element of these vectors in the previous time step.

$$\mathbf{v}_j \leftarrow b\mathbf{v}_{j+1} \quad (16)$$

Since the flyer may rotate about its vertical axis, the body-fixed coordinate system changes relative to an inertial frame. A coordinate system fixed to the visual system rotates through an angle $\Delta\theta$ when moving from one time step to the next. The differential angular displacement $\Delta\theta$ is determined directly from the optic flow field. The rotation matrix, $\mathbf{C}$, that transforms vectors from the reference frame of the previous time step to the reference frame of the current time step is given by (17).

$$\mathbf{C} = \begin{bmatrix} \cos(\Delta\theta) & 0 & -\sin(\Delta\theta) \\ 0 & 1 & 0 \\ \sin(\Delta\theta) & 0 & \cos(\Delta\theta) \end{bmatrix} \quad (17)$$

The short-term memory of air vector measurements contained in the batch air vector $\mathbf{V}_a$ must be updated so that all vectors are represented in the current body-fixed coordinate system. The batch air vector is updated across time steps by using (18), where $\mathbf{v}(t)$ is the latest measurement of the air vector taken at the current time.

$$\mathbf{v}_{a,j} \leftarrow \begin{cases} \mathbf{C}\mathbf{v}_{a,j+1} & 1 \leq j \leq n_{V_a} - 1 \\ \mathbf{v}(t) & j = n_{V_a} \end{cases} \quad (18)$$

In our simplified formulation, we assumed that the apparent egoveloccity, $\boldsymbol{\gamma}$, can be computed directly from the optic flow field. However, the optic flow field actually provides an estimate of the apparent egodisplacement vector, $\boldsymbol{\lambda}$, which may be computed along all three Cartesian axes, for example, using the technique of Nagle and Srinivasan (1996). The apparent egodisplacement is the sum of two distinct components over a short discrete time period. The first component of $\boldsymbol{\lambda}$ comes from the motion of the flyer relative to the

inertial reference frame. The second component of λ comes from the change in ground elevation as the flyer traverses over the ground (Figure 1). If the ground is inclined relative to the viewing axis of the visual system, the distance to the ground will change as the flyer traverses in the x and z directions, even if there is no vertical component of motion. This change will be perceived by the visual system as a change in scene depth, or looming. If the slope of the ground along the direction of flight is positive and the vertical position of the flyer is constant, the flyer will approach the ground and the apparent egodisplacement in the vertical direction will be negative.

Define a point O as the point on the ground directly beneath the flyer at the current time (Figure 1). Let $\mathbf{p}(t)$ denote the position of the flyer at time t relative to point O and represented with respect to the body-fixed coordinate system at the current time. Let $h(t)$ denote the true height of the flyer above the ground, and let $h_g(t)$ denote the height of the ground, relative to a horizontal reference plane passing through point O . Thus, h_g is defined to be zero at the current time step. The apparent egodisplacement can be defined using (19), where $\mathbf{h}_g = [0 \ h_g \ 0]^T$.

$$\begin{aligned}
\lambda(t) &= \frac{1}{h(t)} \left(\frac{\mathbf{p}(t) - \mathbf{p}(t-\Delta t)}{\Delta t} - \frac{\mathbf{h}_g(t) - \mathbf{h}_g(t-\Delta t)}{\Delta t} \right) \\
&= \frac{1}{h(t)\Delta t} (\Delta \mathbf{p}(t) - \Delta \mathbf{h}_g(t)) \\
&= \frac{1}{h(t)\Delta t} \begin{bmatrix} \Delta p_x \\ \Delta p_y - \Delta h_g \\ \Delta p_z \end{bmatrix} \\
&= \frac{1}{h(t)\Delta t} \begin{bmatrix} \Delta p_x \\ \Delta h \\ \Delta p_z \end{bmatrix}
\end{aligned} \tag{19}$$

We store values of λ over a short time period in a batch vector, Λ . If we store the apparent egodisplacement over the batch window as a function only of the current height, it will later only be necessary to solve for the current height rather than solve for the height at each time in the batch window. By rearranging the expression for λ_y in (19), we formulate an estimate of the height at the previous time step given an estimate of the current height and a measurement of the apparent egodisplacement, as in (20).

$$\hat{h}(t - \Delta t) = (1 - \lambda_y(t) \cdot \Delta t) \hat{h}(t) \tag{20}$$

We define a scale factor, s_h , in (21), that represents the relative change in height from one time step to the next.

$$s_h = 1 - \lambda_y(t) \cdot \Delta t \tag{21}$$

Thus, we can update Λ across time steps using (22).

$$\lambda_j \leftarrow s_h \mathbf{C} \lambda_{j+1} \quad 1 \leq j \leq n_\Lambda - 1 \tag{22}$$

We now need to estimate the apparent egovelocity, γ , from the egodisplacement vector. We first define the relative apparent position as the summation of egodisplacement from the start of the batch window up to some point in time. The simplest method of estimating the apparent egovelocity is to use a backward difference scheme applied to the relative apparent position. This is equivalent to setting the egovelocity equal to the egodisplacement. The egovelocity can also be estimated from the egodisplacement using a central difference scheme, whereby the egovelocity is estimated at the middle of three relative apparent position points from the slope of a line fit to those three points. Higher order methods of estimating egovelocity from relative apparent position can also be devised. In a method described by (Lanczos, 1988) and used by (Rayner and Aldridge, 1985), velocity is calculated from the slope of a parabola that is fit to a neighborhood of five sampled position points (or a neighborhood of four points near the ends of the batch window). We will refer to this method as the parabolic difference scheme. One may also choose not to compute finite differences near the endpoints, resulting in a truncated set of egovelocity estimates. The backward, central, central truncated, parabolic, and parabolic truncated egovelocity computation methods are summarized in Table 1.

Table 1: Methods of computing apparent egovelocity, γ , from apparent egodisplacement, λ

method	apparent egovelocity computation	batch vector lengths
backward	$\gamma_{n_\Gamma} = \lambda_{n_\Lambda}$	$n_{V_a} = n$ $n_\Gamma = n$ $n_\Lambda = 1$
central	$\gamma_{n_\Gamma-1} = \frac{1}{2}(\lambda_{n_\Lambda} + \lambda_{n_\Lambda-1})$ $\gamma_{n_\Gamma} = \lambda_{n_\Lambda}$	$n_{V_a} = n$ $n_\Gamma = n$ $n_\Lambda = 2$
central truncated	$\gamma_{n_\Gamma} = \frac{1}{2}(\lambda_{n_\Lambda} + \lambda_{n_\Lambda-1})$	$n_{V_a} = n$ $n_\Gamma = n - 1$ $n_\Lambda = 2$
parabolic	$\gamma_{n_\Gamma-2} = \frac{1}{10}(2\lambda_{n_\Lambda} + 3\lambda_{n_\Lambda-1} + 3\lambda_{n_\Lambda-2} + 2\lambda_{n_\Lambda-3})$ $\gamma_{n_\Gamma-1} = \frac{1}{20}(11\lambda_{n_\Lambda} + 8\lambda_{n_\Lambda-1} + \lambda_{n_\Lambda-2})$ $\gamma_{n_\Gamma} = \frac{1}{20}(21\lambda_{n_\Lambda} + 8\lambda_{n_\Lambda-1} - 9\lambda_{n_\Lambda-2})$	$n_{V_a} = n - 1$ $n_\Gamma = n - 1$ $n_\Lambda = 4$
parabolic truncated	$\gamma_{n_\Gamma} = \frac{1}{10}(2\lambda_{n_\Lambda} + 3\lambda_{n_\Lambda-1} + 3\lambda_{n_\Lambda-2} + 2\lambda_{n_\Lambda-3})$	$n_{V_a} = n - 1$ $n_\Gamma = n - 3$ $n_\Lambda = 4$

The augmented batch vector of apparent egovelocity, $\mathbf{\Gamma}$, is updated from one time step to the next using (23). Notice that it is not necessary to compute the egovelocity from the egodisplacement over each of the n_Γ samples. Instead, only the most recent values of γ need to be computed from λ using one of the apparent egovelocity computation methods listed in Table 1.

$$\gamma_j \leftarrow s_h \mathbf{C} \gamma_{j+1} \quad 1 \leq j \leq n - n_\Lambda \quad (23)$$

2.2.1 Ground shape model

In our simplified formulation, we assumed that the ground was level. We now extend our formulation to allow for flight over an inclined ground. We consider two cases - when the ground slope is known and when it is not known. We define a vector $\hat{\xi}$ that contains the unknown quantities that must be estimated. In the first case, $\hat{\xi}$ represents the unknown height, in the second case, $\hat{\xi}$ is a vector containing estimates of both the height and the ground slope (Table 2).

An estimate of the ground vector can be computed from the apparent egovelocity, an estimate of the ground height, $\hat{\mathbf{H}}_g$, and the estimate of the height of the flyer using (24) as shown in Figure 1.

$$\mathbf{V}_g = \mathbf{\Gamma} h - \frac{\partial}{\partial t} \hat{\mathbf{H}}_g \quad (24)$$

If the ground height is known, $\hat{\mathbf{H}}_g$ is replaced with $\mathbf{H}_g$, otherwise, a model of the ground height must be used. The ground can be modeled as a plane and expressed as a linear function of x and z . Although higher order ground shape models are certainly possible, they are not explored here. Let b_{gx} denote the product of the height of the flyer, h , and the slope of the ground in the x -axis (Figure 1). Similarly, define b_{gz} in the z -axis of the flyer, and let $\mathbf{B}_g = [b_{gx} \ b_{gz}]^T$. The terms b_{gx} and b_{gz} are defined in this way so that they may later be estimated from a linear system of equations. The rate of change in height of the ground as a function of time at the point directly beneath the flyer can be written as in (25) and (26).

$$\begin{aligned} \frac{\partial}{\partial t} h_g &= \frac{\partial h_g}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial h_g}{\partial z} \frac{\partial z}{\partial t} \\ &= \frac{b_{gx}}{h} v_{gx} + \frac{b_{gz}}{h} v_{gz} \end{aligned} \quad (25)$$

$$\frac{\partial}{\partial t} \hat{\mathbf{H}}_{gy} = \frac{\partial}{\partial t} \begin{bmatrix} \hat{h}_g(t - (n_\Gamma - 1)\Delta t) \\ \vdots \\ \hat{h}_g(t - \Delta t) \\ \hat{h}_g(t) \end{bmatrix} = \mathbf{\Gamma}_x \hat{b}_{gx} + \mathbf{\Gamma}_z \hat{b}_{gz} \quad (26)$$

$$\frac{\partial}{\partial t} \hat{\mathbf{H}}_{gx} = \frac{\partial}{\partial t} \hat{\mathbf{H}}_{gz} = \mathbf{0}$$

By substituting (26) into (24) and arranging in matrix form, we obtain the expressions for the estimate of the batch ground vector, $\hat{\mathbf{V}}_g$, given in Table 2.

2.2.2 Wind model

The batch wind vector over n_Γ time steps is estimated using (27), where the truncated batch air vector, $\mathbf{V}'_a$, only includes the first n_Γ terms of each Cartesian component of $\mathbf{V}_a$.

$$\hat{\mathbf{W}} = \hat{\mathbf{V}}_g - \mathbf{V}'_a \quad (27)$$

In our simplified formulation, we assumed that the wind was constant. We now assume that the wind may be modeled by a low order polynomial over the batch window. Let $\check{\mathbf{w}}$ be a polynomial approximation to the true wind vector $\mathbf{w}$. The expression for $\check{\mathbf{w}}$ is given by (28), where m is the degree of the polynomial. The coefficients of the polynomial curve are organized into an augmented batch vector $\mathbf{B}_w$ of length $3(m+1)$ as given by (29).

$$\check{\mathbf{w}}(t - j\Delta t) = \mathbf{b}_{w0} + \mathbf{b}_{w1}(t - j\Delta t) + \cdots + \mathbf{b}_{wm}(t - j\Delta t)^m \quad (28)$$

$$\mathbf{B}_{we_k} = \begin{bmatrix} b_{w0e_k} \\ b_{w1e_k} \\ \vdots \\ b_{wme_k} \end{bmatrix} \quad k = 1, 2, 3 \quad \mathbf{B}_w = \begin{bmatrix} \mathbf{B}_{we_1} \\ \mathbf{B}_{we_2} \\ \mathbf{B}_{we_3} \end{bmatrix} \quad (29)$$

From (28) and (29), we formulate a matrix equation that defines the polynomial approximation to the wind vector over n_Γ time steps. First, we define the Vandermonde matrix $\mathbf{T}$ as in (30), which replaces the vector $\mathbf{1}$ in (9) in the simplified formulation (note that $\mathbf{1}$ is a Vandermonde matrix with $m = 0$).

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & \Delta t & (\Delta t)^2 & \cdots & (\Delta t)^m \\ 1 & 2\Delta t & (2\Delta t)^2 & \cdots & (2\Delta t)^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & (n_\Gamma - 1)\Delta t & ((n_\Gamma - 1)\Delta t)^2 & \cdots & ((n_\Gamma - 1)\Delta t)^m \end{bmatrix} \quad (30)$$

The wind model for each Cartesian axis is now formulated using (31), and the terms $\check{\mathbf{W}}_{e_1}$, $\check{\mathbf{W}}_{e_2}$, and $\check{\mathbf{W}}_{e_3}$ are stacked into an augmented batch vector $\check{\mathbf{W}}$ of length $3n_\Gamma$ using (32).

$$\check{\mathbf{W}}_{e_k} = \mathbf{T}\mathbf{B}_{we_k} \quad k = 1, 2, 3 \quad (31)$$

$$\check{\mathbf{W}} = \begin{bmatrix} \check{\mathbf{W}}_{e_1} \\ \check{\mathbf{W}}_{e_2} \\ \check{\mathbf{W}}_{e_3} \end{bmatrix} = \begin{bmatrix} \mathbf{T} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{T} \end{bmatrix} \mathbf{B}_w \quad (32)$$

Table 2: Formulation of the vector of unknown parameters, $\hat{\xi}$, the batch ground vector, $\hat{\mathbf{V}}_g$, and the coefficient matrix, $\mathbf{L}$

	$\hat{\xi}$	$\hat{\mathbf{V}}_g$	$\mathbf{L}$
ground slope known	$\hat{h}$	$\begin{bmatrix} \Gamma_x \\ \Gamma_y \\ \Gamma_z \end{bmatrix} \hat{h} - \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \Gamma_x & \Gamma_z \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{B}_g$	$\begin{bmatrix} \Theta \Gamma_x \\ \Theta (-b_{gx} \Gamma_x + \Gamma_y - b_{gz} \Gamma_z) \\ \Theta \Gamma_z \end{bmatrix}$
ground slope unknown	$\begin{bmatrix} \hat{h} \\ \hat{\mathbf{B}}_g \end{bmatrix}$	$\begin{bmatrix} \Gamma_x & \mathbf{0} & \mathbf{0} \\ \Gamma_y & -\Gamma_x & -\Gamma_z \\ \Gamma_z & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{h} \\ \hat{b}_{gx} \\ \hat{b}_{gz} \end{bmatrix}$	$\begin{bmatrix} \Theta \Gamma_x & \mathbf{0} & \mathbf{0} \\ \Theta \Gamma_y & -\Theta \Gamma_x & -\Theta \Gamma_z \\ \Theta \Gamma_z & \mathbf{0} & \mathbf{0} \end{bmatrix}$

2.2.3 Height and ground slope estimation

The height and ground slope are estimated by finding values for $\hat{\xi}$ and $\mathbf{B}_w$ that solve the minimization problem of (8). If we assume for the moment that $\hat{\xi}$ is known, then $\hat{\mathbf{V}}_g$ can be found from Table 2, $\hat{\mathbf{W}}$ can be computed from (27), and (8) can be treated as a simple linear least squares problem. The least squares solution for the wind coefficient vector $\mathbf{B}_w$ is given by (33).

$$\mathbf{B}_w = \begin{bmatrix} \mathbf{T} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{T} \end{bmatrix}^+ \hat{\mathbf{W}} \quad (33)$$

Substituting the expression in (33) back into (32), (7), and (8) yields another linear least squares problem for $\hat{\xi}$, where $\mathbf{I}$ is an $n_T \times n_T$ identity matrix.

$$\min_{\hat{\xi}} \left(\begin{bmatrix} (\mathbf{T}\mathbf{T}^+ - \mathbf{I}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\mathbf{T}\mathbf{T}^+ - \mathbf{I}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & (\mathbf{T}\mathbf{T}^+ - \mathbf{I}) \end{bmatrix} \hat{\mathbf{W}} \right) \quad (34)$$

The minimization problem of (34) can be solved efficiently by first taking the reduced QR factorization of $\mathbf{T}$, where $\mathbf{Q}_T$ is an orthonormal matrix with the same dimensions as $\mathbf{T}$, and $\mathbf{R}_T$ is a square upper triangular matrix with dimensions $(m+1) \times (m+1)$. The expression $\mathbf{T}\mathbf{T}^+ - \mathbf{I}$ can now be simplified as follows by taking advantage of the fact that an orthonormal matrix satisfies the property that $\mathbf{Q}_T^T \mathbf{Q}_T = \mathbf{I}$.

$$\mathbf{T}\mathbf{T}^+ - \mathbf{I} = \mathbf{Q}_T \mathbf{Q}_T^T - \mathbf{I} = \Theta \quad (35)$$

Since $\mathbf{T}$ does not depend on time and Δt is constant, the QR factorization of $\mathbf{T}$ only needs to be performed once. Also, using the matrix $\mathbf{Q}_T$ as in (35) requires storage only for the matrix $\mathbf{Q}_T$ rather than separate matrices $\mathbf{T}$ and $\mathbf{T}^+$, and avoids the explicit calculation of the pseudoinverse, which can be numerically unstable.

We may now solve for $\hat{\xi}$. We define a matrix $\mathbf{L}$ representing the coefficient of $\hat{\xi}$ in the minimization problem. By substituting the expressions for $\hat{\mathbf{V}}_g$ and $\mathbf{L}$ from Table 2 into (27), then into (33), the minimization problem takes the form of (36), which may be solved using any linear least-squares solution method.

$$\hat{\xi} = \arg \min_{\hat{\xi}} \left\| \mathbf{L} \hat{\xi} - \begin{bmatrix} \Theta & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Theta \end{bmatrix} \mathbf{V}'_a \right\|^2 \quad (36)$$

Notice that the wind model coefficients in the vector $\mathbf{B}_w$ do not need to be explicitly calculated to determine $\hat{\xi}$.

2.2.4 Ground vector and wind vector estimation

Once the height and ground slopes are estimated, the ground vector at the current time may be estimated using (37), where $\hat{b}_{gx}$ and $\hat{b}_{gz}$ may be replaced with b_{gx} and b_{gz} if the ground slopes are known. However, λ

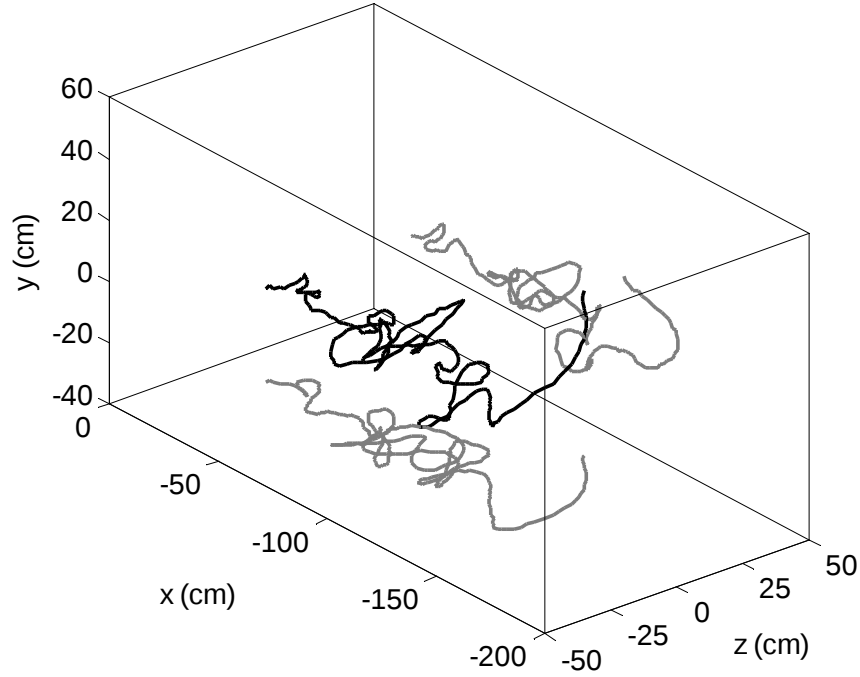


Figure 3: Flight track of a pheromone tracking moth. The moth generally moves from right to left. Projections of the flight track onto horizontal and vertical planes are shown in gray.

is the apparent egodisplacement and may be quite noisy since it comes directly from the measured optic flow. Alternatively, the ground vector may be estimated from the most recent value of the apparent egovelocity, γ , as in (38). However, the most recent egovelocity estimate may not correspond to the current time, depending on the finite differentiation scheme that is used. We will examine the usefulness of both of these estimators in the results section. Once an estimate of the ground vector is obtained, the wind vector is estimated from (1).

$$\hat{\mathbf{v}}_g(t) = \begin{bmatrix} \hat{h}\lambda_{xn_\Lambda} \\ \hat{h}\lambda_{yn_\Lambda} - \hat{b}_{gx}\lambda_{xn_\Lambda} - \hat{b}_{gz}\lambda_{zn_\Lambda} \\ \hat{h}\lambda_{zn_\Lambda} \end{bmatrix} \quad (37)$$

$$\hat{\mathbf{v}}_g(t) = \begin{bmatrix} \hat{h}\gamma_{xn_\Gamma} \\ \hat{h}\gamma_{yn_\Gamma} - \hat{b}_{gx}\gamma_{xn_\Gamma} - \hat{b}_{gz}\gamma_{zn_\Gamma} \\ \hat{h}\gamma_{zn_\Gamma} \end{bmatrix} \quad (38)$$

2.3 Test procedure

Position data of a moth tracking an odor plume in a wind tunnel were used to test the opto-aeronautic algorithm. The data were taken from an experiment in which the moth was recorded with two cameras and the position of both the head and the tail were digitized at 30 Hz. This allowed the body axis orientation to be calculated. The particular trial chosen had 563 data points ($N=563$) or nearly 19 seconds worth of flight data (Figure 3).

Although the moths in the experiment flew in a constant wind, the performance of the opto-aeronautic algorithm was tested with both a constant wind and in natural wind conditions. Natural wind velocity data were recorded parallel to the ground in an open athletic field (Van Horn Field at Case Western Reserve University) using an orthogonal set of acoustic anemometers (Bailey, 2004). It is assumed that no wind

Table 3: Methods of computing the ground vector, $\mathbf{v}_g$, from position data, $\mathbf{p}$

method	$\mathbf{V}_g$
backward	$\mathbf{v}_g(t_j) \approx \begin{cases} \frac{\mathbf{p}_{j+1}-\mathbf{p}_j}{\Delta t} & \text{if } j = 1 \\ \frac{\mathbf{p}_j-\mathbf{p}_{j-1}}{\Delta t} & \text{if } j \geq 2 \end{cases}$
central	$\mathbf{v}_g(t_j) \approx \begin{cases} \frac{\mathbf{p}_{j+1}-\mathbf{p}_j}{\Delta t} & \text{if } j = 1 \\ \frac{\mathbf{p}_{j+1}-\mathbf{p}_{j-1}}{2\Delta t} & \text{if } 2 \leq j \leq N-1 \\ \frac{\mathbf{p}_j-\mathbf{p}_{j-1}}{\Delta t} & \text{if } j = N \end{cases}$
parabolic	$\mathbf{v}_g(t_j) \approx \begin{cases} \frac{-9\mathbf{p}_{j+3}+17\mathbf{p}_{j+2}+13\mathbf{p}_{j+1}-21\mathbf{p}_j}{20\Delta t} & \text{if } j = 1 \\ \frac{\mathbf{p}_{j+2}+7\mathbf{p}_{j+1}+3\mathbf{p}_j-11\mathbf{p}_{j-1}}{20\Delta t} & \text{if } j = 2 \\ \frac{2\mathbf{p}_{j+2}+\mathbf{p}_{j+1}-\mathbf{p}_{j-1}-2\mathbf{p}_{j-2}}{10\Delta t} & \text{if } 3 \leq j \leq N-2 \\ \frac{11\mathbf{p}_{j+1}-3\mathbf{p}_j-7\mathbf{p}_{j-1}-\mathbf{p}_{j-2}}{20\Delta t} & \text{if } j = N-1 \\ \frac{21\mathbf{p}_j-13\mathbf{p}_{j-1}-17\mathbf{p}_{j-2}+9\mathbf{p}_{j-3}}{20\Delta t} & \text{if } j = N \end{cases}$
smoothing spline	GCVSPL implemented in Matlab (Woltring, 1986; Reina, 1998)

existed in the y-direction (perpendicular to the ground). The wind was sampled at 60 Hz with a resolution of approximately 10 cm/s. The wind data were then low pass filtered to remove digitization effects and resampled at 30 Hz.

Ground truth values for the ground vector and air vector were computed from the wind velocity and moth position data using several different methods. The ground vector was calculated using the backward, central, and parabolic ground vector estimation methods in Table 3, and by differentiating a natural cubic smoothing spline fit to the position data using a Matlab implementation of the GCVSPL software package (Woltring, 1986; Reina, 1998). Using these different methods allowed us to detect any anomalies in the performance of the opto-aeronautic algorithm that were an artifact of the ground vector estimation method. The ground vector was then transformed into the body-fixed coordinate system and used as the true ground vector, $\mathbf{v}_g$. The air vector was calculated by subtracting the wind vector from the ground vector. The apparent egodisplacement, $\boldsymbol{\lambda}$, was then calculated using (19), and the rotational angular egodisplacement, $\Delta\theta$, was calculated as the angular difference in the longitudinal axis of the flyer in successive samples multiplied by the sampling frequency of 30 Hz.

The performance of the opto-aeronautic algorithm was examined for both the deterministic (no noise) sensor case and the stochastic sensor case. In the stochastic case, the airdata and apparent egodisplacement were injected with uniformly distributed noise prior to input into our opto-aeronautic algorithm. Since Franz et al. (2004) achieved a mean relative error of 7.5% for computing apparent translational egomotion from optic flow, and the moths in our experiment were flying at a speed of approximately 30 cm/s at a height of roughly 40 cm above the floor, we used an error in the range of ± 0.1 radians per second for the translational component of apparent egomotion. Franz et al. (2004) also achieved a mean error of 0.01 radians per second for rotational motion, which is negligible. We used airdata noise in the range of ± 10 cm/s, which roughly corresponds to the performance of the acoustic anemometers that we used to collect the wind velocity data.

3 Results

The mean absolute error in height, E_h , was used to assess the accuracy of the opto-aeronautic algorithm over the length of the flight track. Since the ground vector and wind vector are calculated from the height estimate, a low value of E_h indicates that good estimates of the ground vector and wind vector were obtained.

Since the algorithm has an initialization period during which the batch measurement vectors of length n are filled, the first n estimates were excluded from the calculation of E_h . For a flight track of N total points, E_h is calculated using (39).

$$E_h = \frac{1}{N - n} \sum_{i=n+1}^N |h(t_i) - \hat{h}(t_i)| \quad (39)$$

3.1 Constant wind condition

In this section, we examine the accuracy of height estimation when the wind is constant in the negative x-direction with a speed of 100 cm/s. We plot the mean absolute error in height as a function of n (the number of points used for the batch window) for each of the egovelocity computation methods given in Table 1 and ground vector computation methods given in Table 3 (where the ground vector computation is used only to generate the inputs of the air vector and egodisplacement vector as described previously). Since the wind is constant, we set m , the degree of the polynomial approximating the wind, to zero.

First, we examine the case when there is no noise in the sensors (Figure 4). With no sensor noise, the backward egovelocity computation method produces a value for E_h on the order of 10^{-14} cm for all values of n when the ground vector is computed using a backward differencing scheme (Figure 4a). The central truncated egovelocity computation method performs similarly when the ground vector is computed using a central differencing scheme (Figure 4b), and the parabolic truncated egovelocity computation method performs similarly when the ground vector is calculated using a parabolic differencing scheme (Figure 4c). This demonstrates that the algorithm performs near the limits of numerical precision when the egovelocity computation method is a perfect model for the ground vector. The non-truncated egovelocity computation methods do not perform as well because they do not perfectly model the ground vector at the endpoints of the batch window. When the ground vector was computed with a smoothing spline, all of the egovelocity computation methods produce errors in the height estimate due to errors in modeling the ground vector (Figure 4d). Even with no sensor noise, it is evident that the backward egovelocity computation method is a bad choice because it performs poorly unless the backward ground vector computation model is a good model for the true ground vector, which is unlikely. Overall, the parabolic and parabolic truncated egovelocity computation methods are quite robust to unmodeled differences between the true ground vectors and their finite difference approximations.

We now separately examine the effects of optic flow and air velocity sensor noise on the performance of the algorithm. The resulting values for E_h with noise only in the optic flow sensors are shown in Figure 5, and Figure 6 shows the results with noise only in the air velocity sensors. Compared to Figure 4, we see that adding a small amount of noise to either sensor produces error in the height estimate, even when the egovelocity computation method matches the ground vector computation method. This is not surprising. If one examines the results closely for the parabolic and parabolic truncated egovelocity computation methods when the ground vector is calculated with either the spline method or the parabolic method, it appears that there is a limit to how well the algorithm can perform as n increases. Overall, the parabolic truncated egovelocity computation method is the best choice since it produces a value of $E_h < 2$ cm for $n \geq 25$ in all tests, and it produces the lowest values for E_h when the most accurate ground vector computation methods are used (the parabolic and smoothing spline methods).

3.2 Natural wind condition

We now examine the performance of the opto-aeronautic algorithm with both noiseless and noisy sensors in the varying natural wind measured at Van Horn Field. Based on our results from tests in a constant wind, we use the parabolic truncated egovelocity computation method and the spline ground vector computation method. To determine the best values to choose for m (the degree of the polynomial approximating the wind) and n (the number of points used for the batch window), the value of m is adjusted from 0 to 4 and the value of n is adjusted from its lowest possible value (depending on the egovelocity computation method used) to 50.

First, we examine the case of flying over level ground. In a varying wind, we see the advantage of using a higher degree polynomial model of the wind (Figure 7). If the wind is variable, the 0th degree polynomial

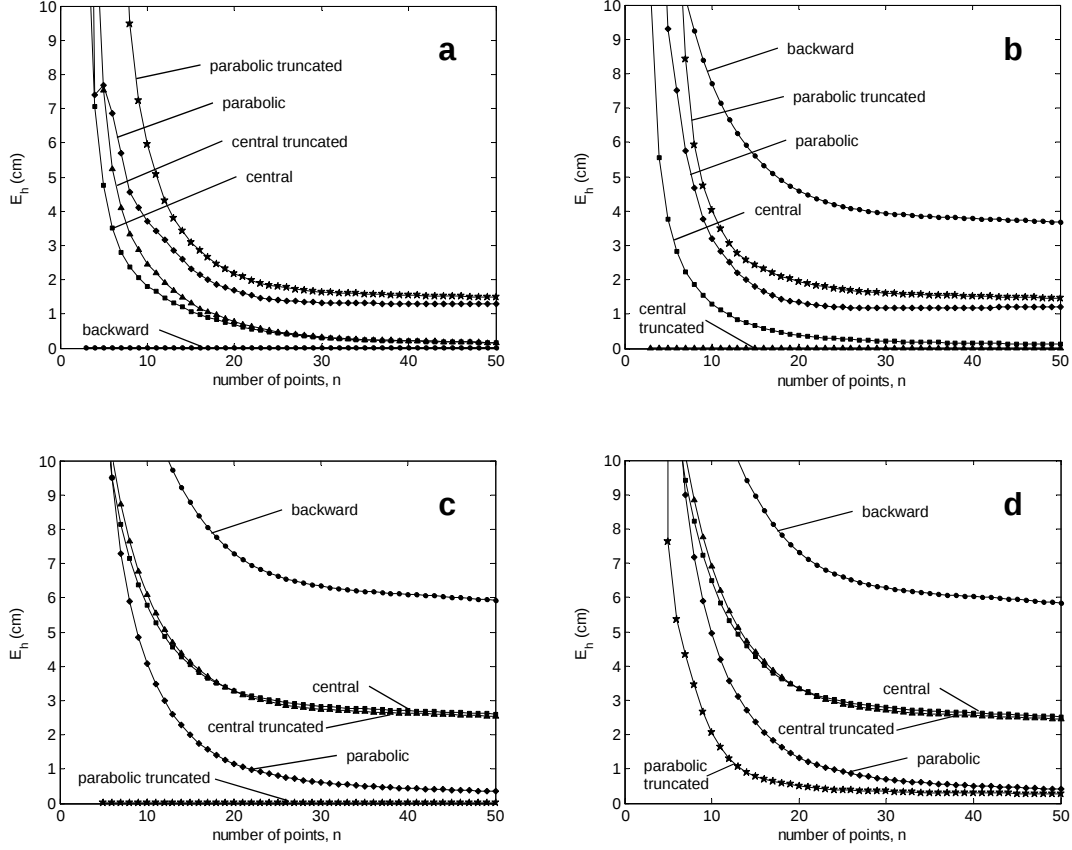


Figure 4: Mean absolute error in the height estimate, E_h , as computed using (39) from the output of the opto-aeronautic egomotion algorithm with zero noise in the air velocity and optic flow input. The results are shown for various values of the batch window length, n , when the ground vector is calculated from the recorded flight track using each of the methods given in Table 3: **a** backward differencing scheme, **b** central differencing scheme, **c** parabolic differencing scheme, **d** smoothing spline. The five different methods listed in Table 1 for estimating egovelocity from measurements of egodisplacement were tested, as indicated by the curve labels.

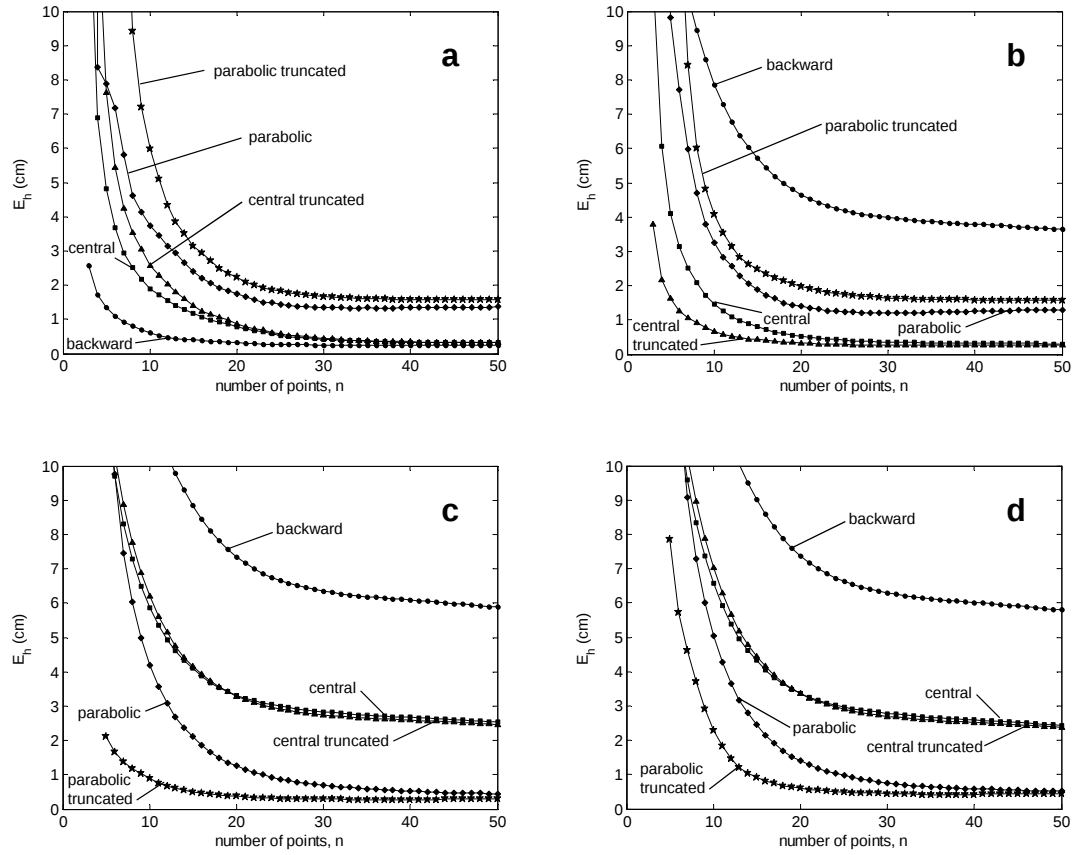


Figure 5: Error in the height estimate, E_h , as computed using the opto-aeronautic egomotion algorithm with uniformly distributed noise of ± 0.1 rad/s in the translational components of optically detected apparent egomotion. Other details are the same as in Figure 4.

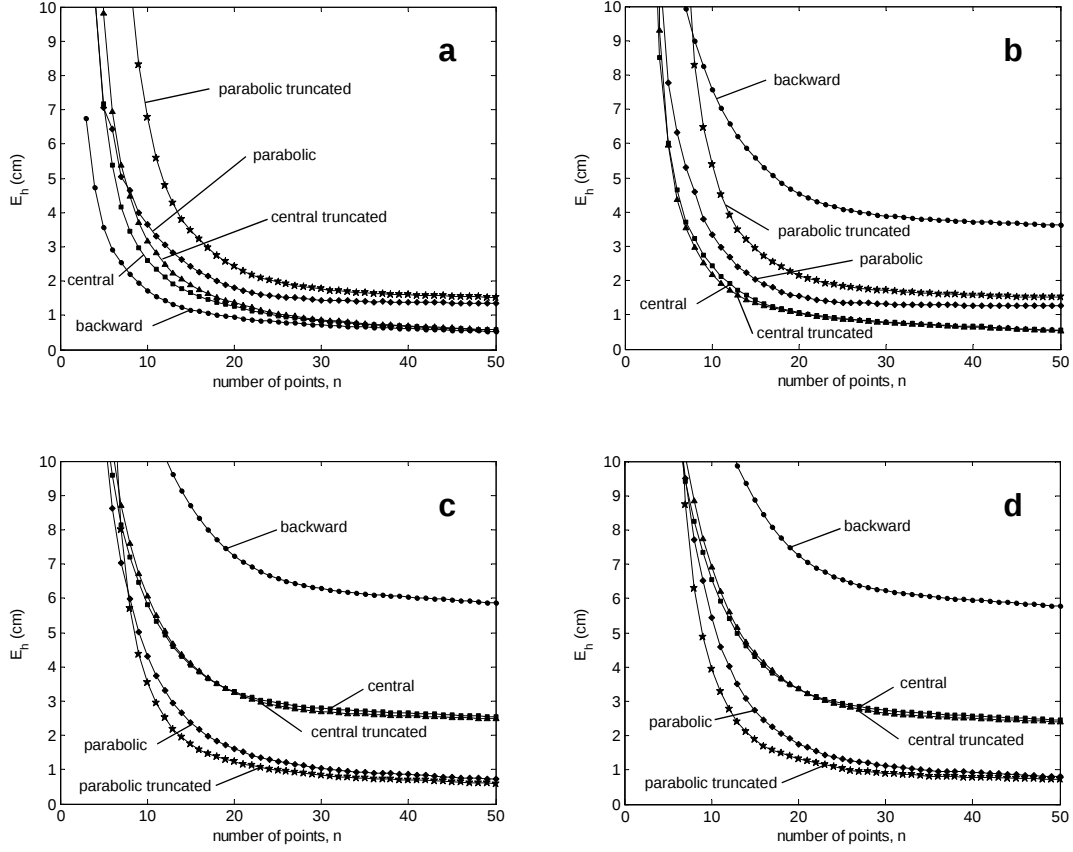


Figure 6: Error in the height estimate, E_h , as computed using the opto-aeronautic egomotion algorithm with uniformly distributed noise of ± 10 cm/s in the air velocity measurements. Other details are the same as in Figure 4.

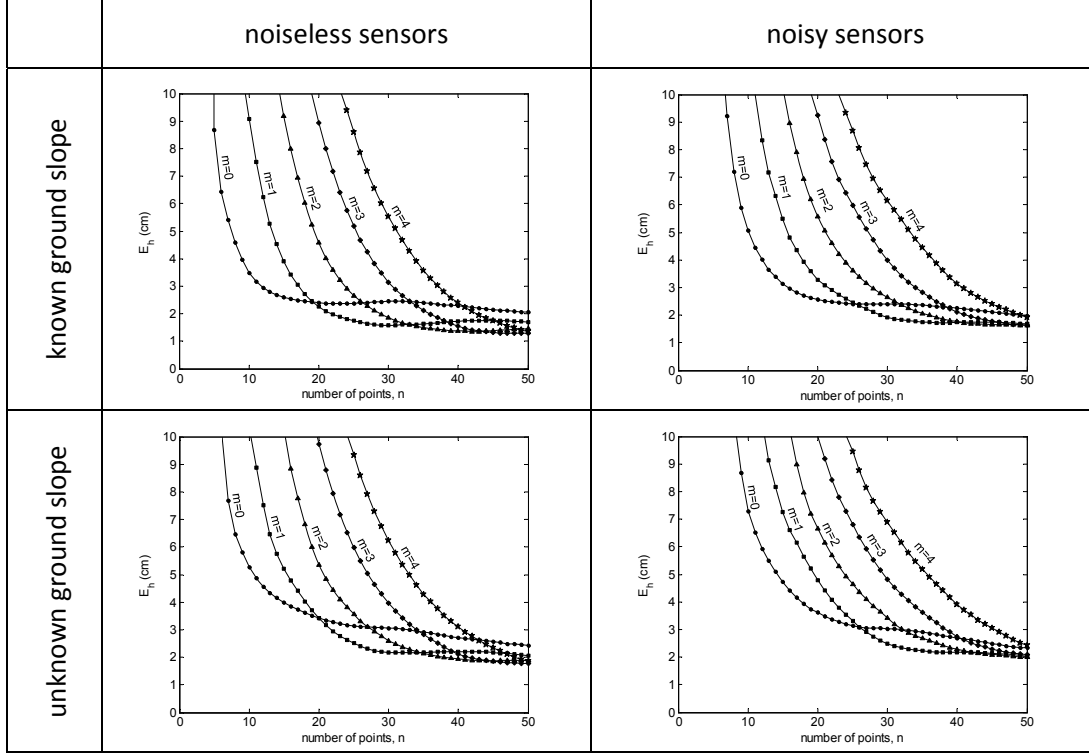


Figure 7: Error in height estimate, E_h , for flight over flat ground in a naturalistic wind environment. The results are shown for various values of the wind model degree, m , and the batch window length, n . The plots on the left show the results for noiseless sensors, while those on the right show the results for noisy sensors. The plots on the top show the performance of the algorithm when the ground slope is known (thus the parameters in the minimization problem of (36) are formulated according to the first row of Table 2), while those on the bottom show the results when the slope is unknown and must be solved for (using the parameters in the second row of Table 2).

performs the best for low values of n , but at higher values of n , the higher degree polynomials perform better. This is reasonable since it is expected that a polynomial of higher degree would be needed to model the wind over a longer time period. In some cases, particularly when $m = 1$, increasing n will actually lead to a slight increase in E_h . Adding noise to the sensors does indeed increase E_h , but only by a small amount for high values of n . Thus, the estimator is fairly robust to sensor noise, and most of the error in estimating the height comes from the inability of the polynomial wind model to perfectly model the true wind profile. An example of the comparison of the height estimate to the true height at each point in the flight track is shown in Figure 8. Notice that in this case, the height estimate never diverges from the true height by more than 11.0 cm.

We now examine the case of flight over an inclined ground with a slope in the x-direction of -0.2 and a slope in the z-direction of 0.2 (Figure 9). The mean absolute error in height only improves slightly when we explicitly solve for the ground slope rather than assume that the ground is level, and only when $n \geq 30$. However, the mean error in height (bias), as computed with (39) but without taking absolute values, is generally reduced when we explicitly solve for ground slope. For example, when $m = 1$ and $n = 50$, the bias in the height estimate goes from 1.8 cm when the ground is assumed level to only 0.1 mm when we explicitly solve for the slope. The mean absolute error in height is even better if the slope of the ground is known, but only by about 0.5 cm or less. Also, the advantage of using a higher degree polynomial approximation to the wind diminishes when noise is added to the sensor data. An example of the comparison of the height and ground slope estimates to their true values at each point in the flight track is shown in Figure 10. The

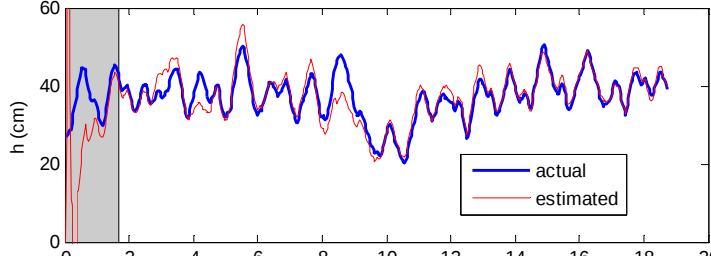


Figure 8: Comparison of the estimated height to the actual height for flight over level ground when $m = 1$, $n = 50$, the ground slope is unknown, and both sensors are corrupted with noise. The batch measurement vectors of length n are filled during the initialization period shown in gray. The mean absolute error in height, E_h , is 2.0 cm.

estimated ground slope has been transformed into the inertial reference frame so that it is easy to compare to the true ground slope. Over the course of the flight track, the mean slope estimates are close to the true values, but the ground slope error has a relatively large variance (slope in x: -0.23 ± 0.12 , slope in z: 0.21 ± 0.05). Thus, the minimization criterion of (36) is not very sensitive to the estimate of ground slope, which makes it difficult to obtain quality slope estimates. A comparison of the two different methods given in (37) and (38) for estimating the ground vector shows that the estimator of (38) provides smoother estimates of the ground vector and the wind vector, but at the expense of a time lag of two time steps, or 0.067 seconds (Figure 11).

4 Discussion

In this work, we have demonstrated that it is possible, while airborne, to estimate absolute egomotion (the ground vector), altitude, ground slope, and the wind vector using only inputs from an onboard downward-looking visual system and onboard air velocity sensors. The method we have described works in both constant and varying wind and even over sloped terrain. Our method also does not require any prior knowledge of the environment or the flyer motion states. This work is significant in two ways. First, it presents a hypothetical method for flying insects to estimate their ground vector and the wind direction. Second, it is a method of velocity estimation that is suitable for aerial vehicles operating in environments where the Global Positioning System (GPS) signal is unavailable.

Although odor tracking insects are clearly able to resolve the wind direction in flight, it is not known exactly how they do this. It has been suggested that odor tracking insects maintain a constant sum of the transverse and longitudinal components of the optic flow vectors in the ventral portion of their visual field (Ludlow, 1984; David, 1986). The wind direction corresponds to the orientation at which the insect detects minimal transverse optic flow, or wind-induced drift. However, a constant sum of transverse and longitudinal optic flow components was not observed in odor tracking *Manduca sexta* (Willis and Arbas, 1998). Furthermore, the mechanism proposed by Ludlow (1984) ignores the relationship among the ground vector, optic flow, and height, and also ignores the role of air velocity sense organs, which are clearly important for insect flight (Budick et al, 2007; Taylor and Krapp, 2008). Input signals from the compound eyes and from hairs that are sensitive to air currents are combined at a very early stage of processing in the insect brain (Taylor and Krapp, 2008). Perhaps this is where the estimation of height and the decomposition of the air vector into ground vector and wind vector components are performed.

GPS is an extremely valuable tool for airborne navigation. However, the GPS signal has limited use in cluttered and indoor environments since the signal may be blocked by structures. The navigation of micro aerial vehicles (MAVs) in an environment without GPS is particularly challenging due to the small size of the vehicle, thus limiting the ability to carry computational power or bulky sensors. There is limited potential for the use of stereoscopically computed range since the performance of a stereo pair of imagers depends on the baseline separation distance between them (the farther apart, the better). Furthermore, since MAVs are more affected by wind disturbances than larger aircraft, the ability to detect the wind could be used to help mitigate its effects on control. Optic flow sensors have already been used on MAVs (Barrows et al., 2003;

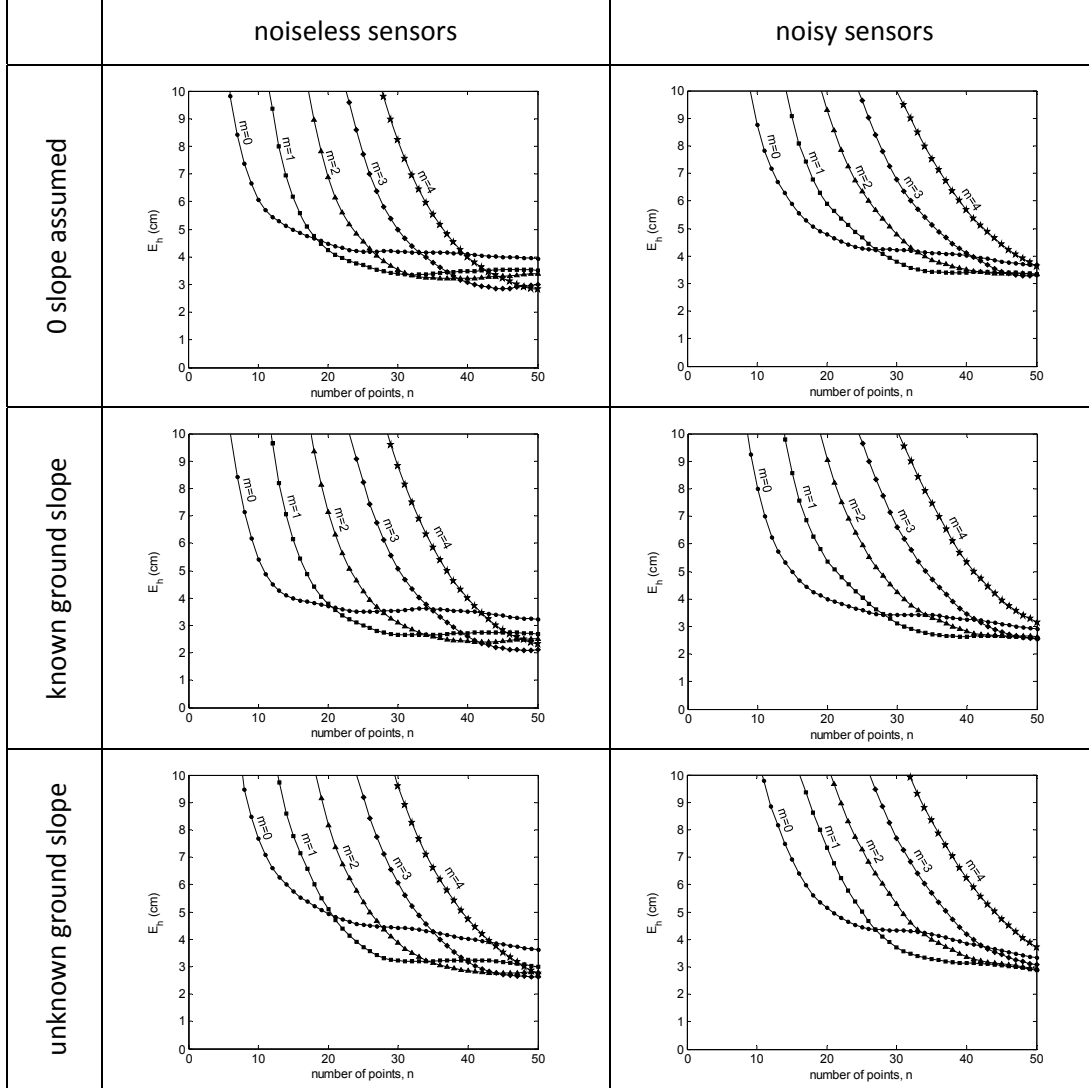


Figure 9: Error in height estimate, E_h , as a function of m and n in a variable wind over a ground with a slope of -0.2 along the x -direction and 0.2 along the z -direction. The plots on the top show the results when the ground is assumed to have 0 slope (by setting $b_{gx} = b_{gz} = 0$ in the parameters of the first row of Table 2), the plots in the middle row show the results when the ground slope is known, and the plots on the bottom show the results when the ground slope is unknown and must be solved for.

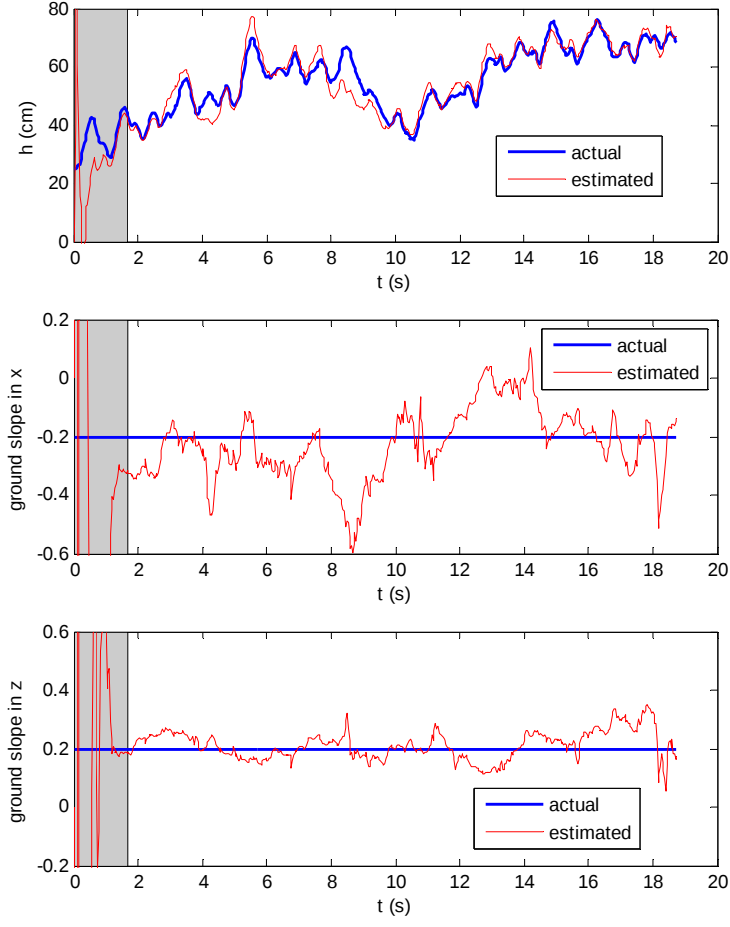


Figure 10: Comparison of the height and ground slope estimates to their true values for flight over an inclined ground when $m = 1$, $n = 50$, the ground slope is unknown, and both sensors are corrupted with noise. The batch measurement vectors of length n are filled during the initialization period shown in gray. The mean absolute error in height, E_h , is 2.9 cm, the estimated ground slope in x has a mean of -0.23 and standard deviation of 0.12 , and the ground slope in z has a mean of 0.21 and standard deviation of 0.05.

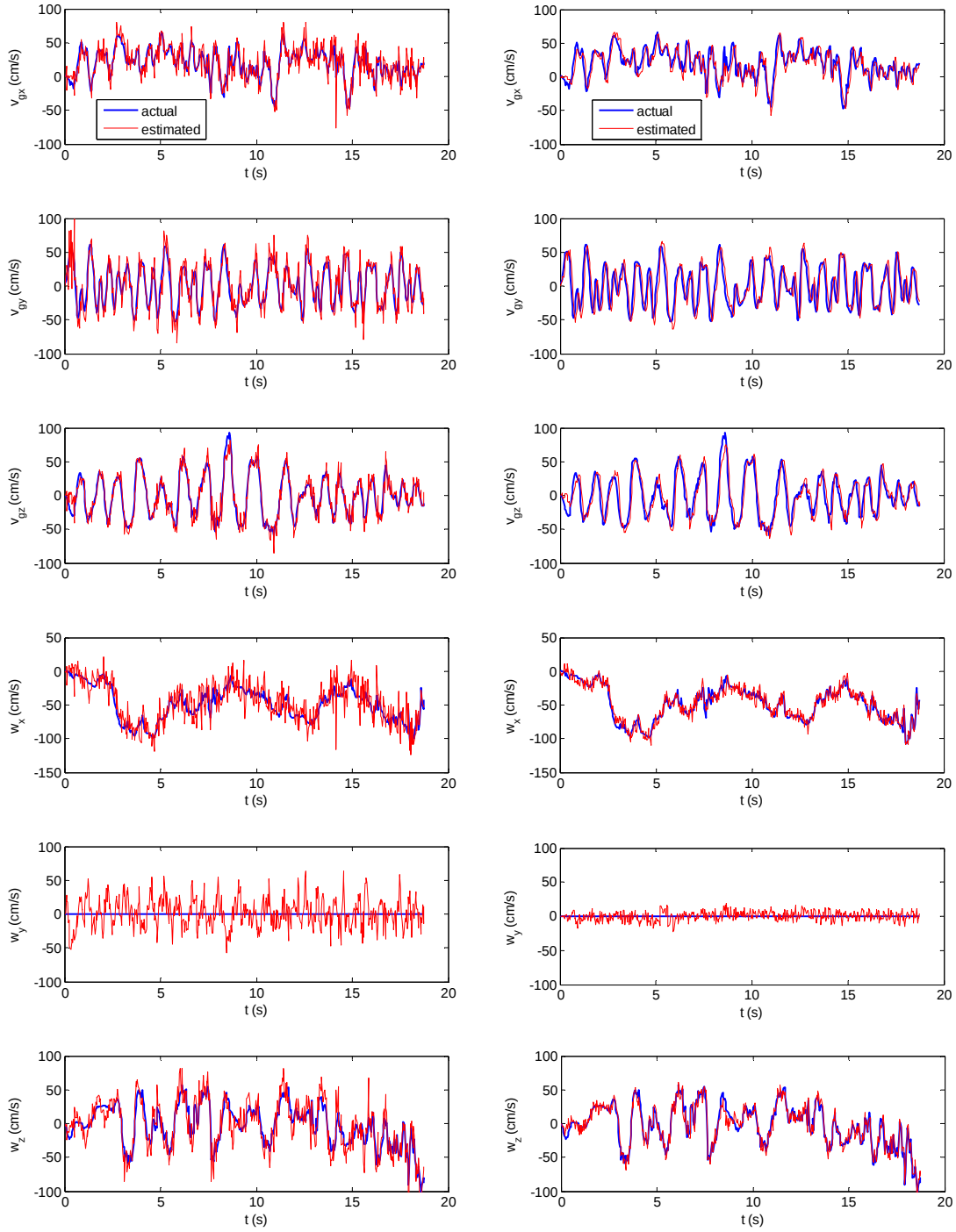


Figure 11: Comparison between the actual and estimated values of the ground vector and wind vector components as computed from the apparent egodisplacement measurement (left, using (37)) and as computed from the apparent egovelocity estimate (right, using (38))

Ruffier and Franceschini, 2004; Zufferey and Floreano, 2005; Griffiths et al., 2006; Conroy et al., 2009), and biomimetic artificial hair cells are a promising technology for air velocity sensing on a small platform (Liu, 2007). Thus, our opto-aeronautic algorithm is particularly well suited to the estimation of egomotion and wind velocity in MAVs.

In the development of our opto-aeronautic algorithm, we restricted our computation of egomotion to use only the downward looking portion of the visual field. Although algorithms exist for computing egomotion from an omnidirectional field of view, the computed egodisplacement vector is either scaled to have unit length (Koenderink and van Doorn, 1987) or requires knowledge of the average scene distance (Franz et al., 2004). An egodisplacement vector of arbitrary scale at each time step is not sufficient for our algorithm, particularly in the computation of the height scale factor, defined in (21), that relates the height at previous time steps to the current height. Instead, the apparent egodisplacement must be computed such that it is scaled with respect to the distance to visual cues, which is possible when the computation of egomotion is restricted to a portion of the visual field with minimal variation in distance.

Since we intend for our opto-aeronautic algorithm to be implemented on an MAV, the algorithm must be computationally efficient so that it can execute on a processing unit with low mass and low power demand. To test if our algorithm is suitable for such an application, we implemented our algorithm on a 16-bit microcontroller (dsPIC30F series) available from Microchip Technologies Inc. With $m = 1$ (which assumes the wind is linear over the batch window) and without the estimation of ground slope, our algorithm could operate at a rate of 60 Hz.

Although we have ignored the contribution of roll and pitch on optic flow, this can be corrected for through the sensing of angular rates. Many flying insects possess structures for measuring their angular velocity. Flies use halteres, which are small club shaped structures that beat at the same frequency as the wings (Pringle, 1948). Recent work indicates that moths may use their antennae to sense angular rates (Sane et al., 2007). These structures would allow a flyer to resolve the ambiguity of determining rotation and translation from the optic flow field.

Although we have demonstrated that our opto-aeronautic algorithm provides good estimates of the height, ground vector, and wind vector, we have not yet developed a method of estimating the accuracy of the outputs of the algorithm without knowledge of the true state. This will be necessary since higher level functionality may depend on how well the egomotion is known. Also, we plan to test the performance of the algorithm with real optic flow and air velocity sensing hardware on a real flying platform.

Acknowledgements

We thank Elizabeth Morrison for providing the digitized moth flight track used to test the algorithm. We thank Justin Bailey for his assistance in measuring the wind velocity in an outdoor environment. We also thank Dr. Benjamin Dickinson for his critical comments on a draft of this manuscript and for his assistance with L^AT_EX. A Rutkowski was supported by the Ohio Space Grant Consortium and the National Research Council.

References

- Bab-Hadiashar A, Suter D, Jarvis R (1995) Image-interpolation based optic flow technique. In: Proceedings of the SPIE Conference on Applications of Digital Image Processing XVIII, pp 271–281
- Barron JL, Fleet DJ, Beauchemin, SS (1994) Performance of optical flow techniques. International Journal of Computer Vision 12:43–77
- Barrows GL, Chahl JS, Srinivasan MV (2003) Biologically inspired visual sensing and flight control. The Aeronautical Journal 107:159–168
- Budick SA, Reiser MB, Dickinson MH (2007) The role of visual and mechanosensory cues in structuring forward flight in *Drosophila melanogaster*. J Exp Biol 210:4092–4103

- Çelik K, Chung S-J, Clausman M, Somani A (2009) Biologically inspired monocular vision based navigation and mapping in GPS-denied environments. In: Proceedings of the AIAA Infotech@Aerospace Conference, pp 1–17.
- Comeaux J (1983) Air navigation. Air Force Manual 51-40.
- Conroy J, Gremillion G, Ranganathan B, Humbert JS (2009) Implementation of wide-field integration of optic flow for autonomous quadrotor navigation. *Autonomous Robots* 27:189–198
- Dahmen HJ, Franz MO, Krapp HG (2001) Extracting egomotion from optic flow: limits of accuracy and neural matched filters. In: Zanker JM and Zeil J (eds) *Motion vision - computational, neural, and ecological constraints*. Springer Verlag, Berlin Heidelberg New York, pp 143–168
- David CT (1986) Mechanisms of directional flight in wind. In: Payne TL, Birch MC, Kennedy CEJ (eds) *Mechanisms in insect olfaction*. Clarendon Press, New York, pp 49–57
- Fadamiro HY, Wyatt TD, Birch MC (1998) Flying beetles respond as moths predict: Optomotor anemotaxis to pheromone plumes at different heights. *J Insect Behav* 11:549–556
- Foster SP, Howard AJ (1999) The effects of source dosage, flight altitude, wind speed, and ground pattern on the sex pheromone-mediated flight manoeuvres of male lightbrown apple moth, *Epiphyas postvittana* (Walker). *New Zeal J of Zool* 26:97–104
- Franz MO, Chahl JS, Krapp HG (2004) Insect-inspired estimation of egomotion. *Neural Comput* 16:2245–2260
- Garratt, MA, Chahl, JS (2008) Vision-based terrain following for an unmanned rotorcraft *Journal of Field Robotics* 25:284–301
- Gewecke M (1974) The antennae of insects as air-current sense organs and their relationship to the control of flight. In: Browne LB (ed) *Experimental analysis of insect behavior*. Springer-Verlag, Berlin, pp 100–113
- Griffiths S, Saunders J, Curtis A, Barber DB, McLain TW, Beard RB (2006) Maximizing miniature aerial vehicles. *IEEE Robotics and Automation Magazine* 13:34–43
- Kennedy JS (1940) The visual responses of flying mosquitoes. *Proc Zool Soc Lond A* 109:221–242
- Kennedy JS, Marsh D (1974) Pheromone regulated anemotaxis in flying moths. *Science* 184:999–1001
- Koenderink JJ, van Doorn AJ (1987) Facts on optic flow. *Biol Cyber* 56:247–254
- Kuenen LPS, Baker TC (1982) Optomotor regulation of ground velocity in moths during flight to sex pheromone at different heights. *Physiol Entomol* 7:193–202
- Lanczos C (1988) *Applied analysis*. Dover Publications
- Liu C (2007) Micromachined biomimetic artificial haircell sensors. *Bioinspiration & Biomimetics* 2:S162–S169
- Ludlow AR (1984) Application of computer modelling to behavioral coordination. Dissertation, University of London
- Marsh D, Kennedy JS, Ludlow AR (1978) An analysis of anemotactic zigzagging flight in male moths stimulated by pheromone. *Physiol Entomol* 3:221–240
- Miao AX, Zacharias GL, Warren R (1996) Passive navigation from image sequences - a practitioner's approach. In: Proceedings of the AIAA Flight Simulation Technologies Conference, pp 726-740
- Nagle MG, Srinivasan MV (1996) Structure from motion: determining the range and orientation of surfaces by image interpolation. *J Opt Soc Am A* 13:25–34
- Nagle MG, Srinivasan MV, Wilson DL (1997) Image interpolation technique for measurement of egomotion in 6 degrees of freedom. *J Opt Soc Am A* 14: 3233-3241.

- Pringle JWS (1948) The gyroscopic mechanism of the halteres of Diptera. *Philos T Roy Soc B* 233:347–384
- Rayner JMV, Aldridge HDJN (1985) Three-dimensional reconstruction of animal flight paths and the turning flight of microchiropteran bats. *J Exp Biol* 118:247–265
- Reina T (1998) Source code for Matlab 5.1 MEX interface for GCVSPL. http://isbweb.org/c/isb/pub/files/orig_website/software/sigproc/gcvspl/reina/source.html
- Ruffier E, Franceschini N (2004) Visually guided micro-aerial vehicle: automatic take off, terrain following, landing and wind reaction. In: *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, pp 2339–2346
- Sane SP, Dieudonné A, Willis MA, Daniel TL (2007) Antennal mechanosensors mediate flight control in moths. *Science* 315:863–866
- Srinivasan MV (1994) An image-interpolation technique for the computation of optical flow and egomotion. *Biol Cybern* 71:401–415
- Srinivasan MV, Poteser M, Kral K (1999) Motion detection in insect orientation and navigation. *Vision Res* 39:2749–2766
- Taylor G, Krapp HG (2008) Sensory systems and flight stability: What do insects measure and why? In: Casas J, Simpson SJ (eds) *Advances in Insect Physiology* 34. Academic Press, pp 231–316
- Toupet, O, Paduano, JD, Panish, R, Sedwick, R, Frazzoli, E Augmenting state estimates with multiple camera visual measurements In: *Proceedings of the AIAA Infotech@Aerospace Conference and Exhibit*.
- Willis MA, Arbas EA (1991) Odor-modulated upwind flight of the sphinx moth, *Manduca sexta* L. *J Comp Physiol A* 169:427–440
- Willis MA, Arbas EA (1998) Variability in odor-modulated flight by moths. *J Comp Physiol A* 182:191–202
- Woltring HJ (1986) A FORTRAN package for generalized, cross-validatory spline smoothing and differentiation. *Adv Eng Softw* 8:104–113
- Zanen PO, Cardé RT (1996) Effects of host-odour plume altitude and changing wind velocity on upwind flight manoeuvres of a specialist braconid parasitoid. *Physiol Entomol* 21:329–338
- Zufferey J-C, Floreano D (2005) Toward 30-gram autonomous indoor aircraft: Vision-based obstacle avoidance and altitude control. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp 2594–2599

DISTRIBUTION LIST
AFRL-RW-EG-TR-2012-091

DEFENSE TECHNICAL INFORMATION CENTER - 1 Electronic Copy (1 file & 1 format)
ATTN: DTIC-OCA (ACQUISITION)
8725 JOHN J. KINGMAN ROAD, SUITE 0944
FT. BELVOIR VA 22060-6218

AFRL/RWOC (STINFO Office)	- 1 Hard (color) Copy
AFRL/RW CA-N	- STINFO Officer to provide notice of publication