On the Computation of the Ego-Motion and Distance to Obstacles for a Micro Air Vehicle

Ram V. Iyer, Zhihai He and Phillip R. Chandler

***Abstract*— In this paper, we have considered the problem of velocity and range estimation for an UAV using a camera and the knowledge of linear speed through a GPS device. In earlier work [1], a method for decomposition of a scene into structure blocks, and finding correspondences between such blocks in successive frames was developed. The result of this low-level image processing is (a) a set of structure blocks; (b) the motion of each structure block; and (c) a reliability index between zero and one denoting the confidence of the solution in (b). Here, we show that by solving a constrained optimization problem set up using the results of the image processing, one can obtain the linear and angular velocity of the camera motion, provided the speed of the linear motion is known. Once the velocity parameters is computed, we show how the range to objects in the field of view can be computed.**

I. INTRODUCTION

In this paper, we have considered the problem of velocity and range estimation for an UAV using vision based techniques. Such a problem is of great importance to Micro Air Vehicles (MAVs) that fly at low enough altitudes so that GPS geo-registration errors can cause them to fly into obstacles. Loss of GPS during flight for short periods of time could also result in the loss a MAV. For such applications such as search and classification of targets, it is necessary for MAV's to carry an on-board camera that streams video signals to a stationary receiver. The question that naturally arises is whether the video data can also help the MAV navigate in the presence of obstacles. Recent work along these lines can be found in [2], [3], [4].

The problem can be broken down naturally into three subproblems:

1) Estimation of motion flow field in the image plane of the camera in an unconstrained environment (that is, the camera is not made to move in a controlled manner);
2) Estimation of the linear and angular velocities of the MAV and the current range of the objects in the visual field; and
3) Navigation to the desired way-points while avoiding the obstacles that might cause the loss of the MAV.

In [1] the first problem of motion flow field estimation is tackled using a correspondence computation scheme that applies to successive frames of a video stream. The result of this image processing is a set of structure blocks that are important features in the scene; a set of motion vectors for these blocks - the "optical flow" field; and a reliability index for each of the structure blocks that indicates the confidence level in the optical flow computation. For example, corner blocks would have a higher reliability index than edge blocks, which in turn would have a higher reliability index than interior blocks. In this paper, we address the second problem listed above - that of estimation of velocities and range to objects. The fundamental question for this sub-problem is whether the linear and angular velocities of the MAV and the current range of the objects in the visual field of an MAV can be computed correctly, assuming that the first question has been solved correctly. The classical solution to the problem is the continuous-time eight point algorithm [5], while we provide an alternate solution methodology in this paper. Assuming that component of the linear velocity along the axis of the camera is positive (this is equivalent to the positive range assumption employed in the literature [5] and is necessary to resolve an ambiguity in the direction of the linear velocity), we show here that subproblem 2 is solvable correctly, provided additional information on the speed of the MAV.

We have considered the case of the calibrated camera in this paper. Our approach is different from the classical continuous eight-point algorithm [5] that is used to determine the linear and angular velocities of the camera. The key difference is that in our approach only a single constrained optimization is required (see Theorem 3.1), while in the continuous eight-point algorithm there are a sequence of four steps that need to be implemented:

1) solution of a constrained linear least squares problem to estimate the continuous essential matrix;
2) recovery of the symmetric epipolar constraint;
3) recovery of the linear velocity; and
4) recovery of the angular velocity from the continuous essential matrix.

R. V. Iyer is with the Department of Mathematics and Statistics, Texas Tech University, Lubbock, TX 79409-1042.

Z. He is with the Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO, 65211.

P. Chandler is with the U.S. Air Force Research Laboratory, Wright-Patterson Air Force Base, Ohio 45433-7531.

There are variations in the continuous eight-point algorithm but all the versions are based on the epipolar constraint equation and hence are based on recovering the essential matrix [5], [14]. Our algorithm side-steps the essential matrix, and directly solves for the linear and angular velocities. Due to availability of high quality feasible sequential quadratic codes, the new approach would be easier to implement on a MAV. In related work, it was shown that in the *absence of the speed information*, the linear and angular velocities could still be estimated provided the camera resembled the compound eye of a house fly [6]. It is clear that the scene being imaged must be “rich” in some sense for the problem to have a solution. In the literature, the the structure blocks are required to be in “general position” [5]. Here we precisely define the notion of a set of *nonsingular structure blocks* that is necessary for the solution of the ego-motion problem.

Going back to the motion field computation problem [1], other techniques for the computation of the flow field on the image plane include the optical flow [7], [8], [3] and scale-invariant feature tracking methods (see [9] and references therein). As the optical flow computation can result in wildly inaccurate solutions (see [7] for a discussion), and in light of Theorem 3.1 it seems that a feature tracking method in some sense is necessary. As there can be translation, rotation and scaling of the image from one frame to the next, it is clear that a scale-invariant approach will be more fruitful. The theoretical results of this paper do not depend on which of the specific motion-field computation methods are used, though we use the notation and terminology employed in [1].

II. Kinematics of Camera Motion

There are several reference frames employed in air vehicle computations. The main ones are the inertial frame and the body frame which is centered on the center of mass. When a camera is used on a UAV, one introduces an additional frame that is centered on the focus of the camera. If the camera is fixed, then the change of coordinates from the body frame to the camera-centered frame is accomplished by a fixed rotation and translation. This is the case we assume in this note.

Figure 1 shows an inertial frame with origin at the point O_i, an UAV body frame with origin at the point O_b and the camera-centered frame origin at the point O_c. For the inertial and UAV body frames the standard convention for labelling the axes is assumed [10] with X_b pointing through the nose of the aircraft, Y_b pointing out the right wing, and Z_b pointing down. For the camera frame, the standard convention used in the machine vision literature is assumed [7] with Z pointing normal

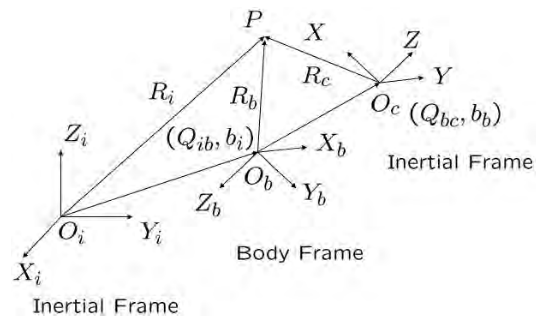

Fig. 1. Kinematics of Camera Motion

to the image plane. Thus if the camera is mounted in front of the aircraft with the image plane parallel to the $Y_b - Z_b$ plane of the aircraft, then X_b and Z will be parallel and perhaps collinear.

The point O_b has coordinates denoted by b_i in the inertial frame. The point O_c has coordinates denoted by b_b in the body frame. Suppose that a point P in space has the inertial coordinates R_i, body coordinates R_b and camera-centered coordinates R_c, then the relation between these coordinates are given by:

$$R_i = Q_{ib} R_b + b_i, \quad (1)$$
$$R_b = Q_{bc} R_c + b_b \quad (2)$$

where $Q_{ib}, Q_{bc} \in SO(3)$ are 3×3 matrices satisfying:

$$Q_{ib}^T Q_{ib} = I; \text{ and } \det(Q_{ib}) = 1 \quad (3)$$
$$Q_{bc}^T Q_{bc} = I; \text{ and } \det(Q_{bc}) = 1 \quad (4)$$

In the following, all variables are assumed to be functions of time unless explicitly stated as constants. Equations (1-2) immediately imply the well-known equations:

$$\dot{Q}_{ib} = Q_{ib} \hat{\Omega}_b, \quad (5)$$
$$\dot{Q}_{bc} = Q_{bc} \hat{\Omega}_c, \quad (6)$$

where $\hat{\Omega}_b$ and $\hat{\Omega}_c$ are skew-symmetric angular velocity matrices, with the subscript indicating the frame where it is defined. Differentiating Equations (1-2) with the condition that the point P is fixed in the inertial frame, we get:

$$\begin{aligned} 0 &= \dot{Q}_{ib} R_b + Q_{ib} \dot{R}_b + \dot{b}_i \quad \text{which implies} \\ \dot{R}_b &= -(Q_{ib}^T \dot{Q}_{ib} R_b + Q_{ib}^T \dot{b}_i) = -(\hat{\Omega}_b R_b + V_b) \ \& \\ \dot{R}_b &= \dot{Q}_{bc} R_c + Q_{bc} \dot{R}_c + \dot{b}_b \quad \text{which implies} \\ \dot{R}_c &= -(\hat{\Omega}_c R_c + V_c) - Q_{bc}^T (\hat{\Omega}_b R_b + V_b) \end{aligned} \quad (7)$$

where: $V_b = Q_{ib}^T \dot{b}_i$ is the linear velocity of the UAV in the body coordinates, and $V_c = Q_{bc}^T \dot{b}_b$ is the linear velocity of the camera in the camera-centered coordinates. If $\Omega_k = [\Omega_{k1}\ \Omega_{k2}\ \Omega_{k3}]^T$ where $k = b$ or c, then $\hat{\Omega}_k$ is a skew-symmetric matrix that satisfies: $\hat{\Omega}_k R_k = \Omega_k \times R_k$.

In special cases, Equation (7) can be simplified:

1) If the camera is fixed on the aircraft, then $\dot{b}_b = 0 \Rightarrow V_c = 0$ and $\dot{Q}_{bc} = 0 \Rightarrow \Omega_c = 0$. This leads

to Equation (7) being modified to:

$$\dot{R}_c = -Q_{bc}^T (\hat{\Omega}_b R_b + V_b).$$

Substituting for R_b from Equation (2), we get:

$$\begin{aligned}\dot{R}_c &= -Q_{bc}^T \hat{\Omega}_b Q_{bc} R_c - Q_{bc}^T \hat{\Omega}_b b_b - Q_{bc}^T V_b \\ &= -\widehat{Q_{bc}\Omega_b} R_c - Q_{bc}^T \hat{\Omega}_b b_b - Q_{bc}^T V_b. \quad (8)\end{aligned}$$

Equation (8) relates the velocity of a point in space in camera-fixed coordinates to linear and angular velocities of the UAV. This is the key equation that will be used in Section III. Notice that even if the camera is positioned so that the normal to the image plane points along the nose of the UAV, the matrix Q_{bc} is given by:

$$Q_{bc} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}. \quad (9)$$

2) In case the camera is gimballed and it is possible to control the angular rate of the camera with b_b fixed, we get:

$$\begin{aligned} u &= \Omega_c \\ \dot{R}_c &= -\hat{u} R_c - \widehat{Q_{bc}\Omega_b} R_c - Q_{bc}^T \hat{\Omega}_b b_b - Q_{bc}^T V_b. \end{aligned}$$

III. Computation of Motion Parameters from Reliability Indexed Motion Field

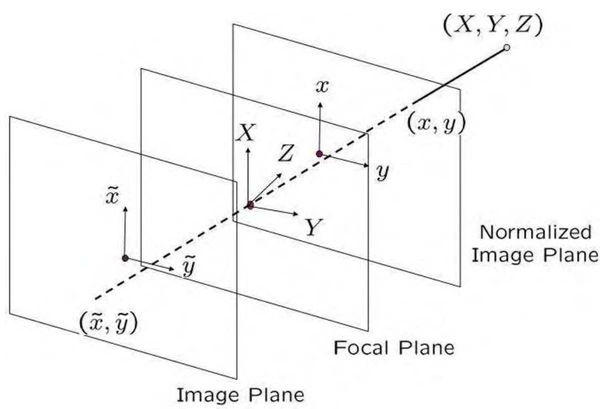

Fig. 2. Notation for the Real Motion Field Computation

Figure 2 shows the notation used for modeling the camera motion. The focal plane contains the focus of the camera and is parallel to the image plane. The origin for the 3 dimensional coordinate system (X, Y, Z) (henceforth referred to as the camera-centered coordinate system) is at the focus of the camera. As is customary in the machine vision literature, it is assumed that the Z axis is normal to the focal plane. The origin for the coordinate system $(\tilde{x}, \tilde{y})$ on the image plane is located at the point $(0, 0, -f)$ in (X, Y, Z) coordinates, where f is focal length of the camera. The point R with coordinates (X, Y, Z) maps to the point $(\tilde{x}, \tilde{y})$ on the image plane with

$$\tilde{x} = -f \frac{X}{Z}, \text{ and } \tilde{y} = -f \frac{Y}{Z}.$$

As the image is inverted, we will consider a "normalized" image plane located at $(0, 0, 1)$ and parallel to the focal plane.

Suppose the point $R_c = (X, Y, Z)$ maps to the point (x, y) on the normalized image plane. Then:

$$x = \frac{X}{Z}, \text{ and } y = \frac{Y}{Z}. \quad (10)$$

If the camera moves with linear velocity $V = (V_X, V_Y, V_Z)$ and angular velocity $\Omega = (\Omega_X, \Omega_Y, \Omega_Z)$, then in the camera-centered coordinate system, the velocity of the point R_c is given by Equation (8):

$$(\dot{X}, \dot{Y}, \dot{Z}) = \widehat{Q_{bc}\Omega_b} R_c - Q_{bc}^T \hat{\Omega}_b b_b - Q_{bc}^T V_b. \quad (11)$$

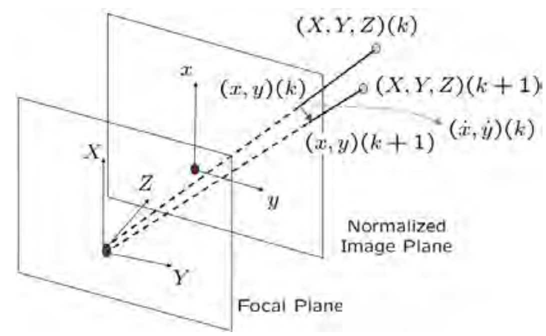

Fig. 3. Object motion in camera frame.

Let $Q_{bc} = [Q_{bc1}\ Q_{bc2}\ Q_{bc3}]$ where we have explicitly written the columns of Q_{bc}. Denote the angular velocity Ω_b represented in the camera-centered frame as $\Omega_c = Q_{bc}\Omega_b \triangleq [\Omega_X\ \Omega_Y\ \Omega_Z]^T$. Then:

$$\begin{aligned}\dot{x} &= \frac{\dot{X}}{Z} - \frac{X\dot{Z}}{Z^2} \\ &\approx \frac{1}{Z}\left(x\langle Q_{bc3}, V_b\rangle - \langle Q_{bc1}, V_b\rangle\right) \\ &\quad + \Omega_X x y - \Omega_Y(1 + x^2) + \Omega_Z y \quad (12)\end{aligned}$$

$$\begin{aligned}\dot{y} &= \frac{\dot{Y}}{Z} - \frac{Y\dot{Z}}{Z^2} \\ &\approx \frac{1}{Z}\left(y\langle Q_{bc3}, V_b\rangle - \langle Q_{bc2}, V_b\rangle\right) \\ &\quad + \Omega_X(1 + y^2) - \Omega_Y x y - \Omega_Z x. \quad (13)\end{aligned}$$

The approximations would be correct if $\frac{b_b}{Z} \approx [0\ 0\ 0]^T$. *Henceforth we will assume this approximation because the distance from the center of mass of the aircraft to the focus of the camera should be much smaller than the distance from the focus to external objects close the axis of the camera.* Note that there might be objects with very small Z value on the periphery of the image, but these objects are not of much interest as they are not obstacles the aircraft needs to avoid.

Let $V_b = [V_{b1}\ V_{b2}\ V_{b3}]^T$. For the special orientation of the camera given by (9), we get (assuming $V_{b1} > 0$ which means that the air vehicle has a positive speed along the nose of the air vehicle):

$$\dot{x} = \frac{V_{b1}}{Z}\left(x + \frac{V_{b3}}{V_{b1}}\right) + \Omega_X xy - \Omega_Y(1+x^2) + \Omega_Z y \quad (14)$$

$$\dot{y} = \frac{V_{b1}}{Z}\left(y - \frac{V_{b2}}{V_{b1}}\right) + \Omega_X(1+y^2) - \Omega_Y xy - \Omega_Z x. \quad (15)$$

An erroneous form of these equations appear in [11]. The correct form with a different method of proof appears in [12].

A. Motion Parameter Computation

We will consider two different methods for motion computation. The time interval $[0, T]$ is partitioned into $0 = T_0 < \cdots < T_k < \cdots < T_N = T$. One assumes that linear velocity at time T_{k-1} is known in the body frame and the angular velocity value at time T_{k-1} is computed at time T_k using the images at times T_{k-1} and T_k. In the second method, both linear and angular velocities are computed simultaneously for time T_{k-1}. As discussed in [1], the image is partitioned into structure and non-structure blocks, $\mathbf{B}^m$, $1 \le m \le P$, and the optical motion vector $(\dot{x}^m, \dot{y}^m)$ is computed for the structure blocks located at (x^m, y^m) with say, $m = 1, \cdots, M$. Let $I : \mathbb{R}^2 \times \mathbb{R}_+ \to \mathbb{R}$ denote the image function. *We will make the standing assumption that if $\mathcal{T}$ denotes the time instants t such that there exists g in the group $SE(2)$ with $I(\cdot,\cdot,t) = I(g(\cdot,\cdot),t)$ then $\mathcal{T}$ has measure zero. Here $g(\cdot,\cdot)$ is the standard action of the group $SE(2)$ on* $\mathbb{R}^2$. This assumption ensures that the aperture problem happens only on a set of measure zero in time.

The variable Z can be eliminated in Equations (14) and (15) we get $P(\Omega, V_b) = 0$, where:

$$P(\Omega, V_b) = V_b^T\left(A(x,y)\,\Omega + B(x,y)\begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix}\right), \quad (16)$$

where

$$A(x,y) = \begin{bmatrix} -x & -y & x^2+y^2 \\ -xy & 1+x^2 & -y \\ -(1+y^2) & xy & x \end{bmatrix}; \quad (17)$$

$$B(x,y) = \begin{bmatrix} -y & x \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (18)$$

To solve for the motion parameters (Ω^*, V_b^*), we find the arguments such that:

$$\begin{aligned}(\Omega^*, V_b^*) &= \underset{\substack{(\Omega, V_b) \\ V_{b1}>0;\ \|V_b\| = V}}{\arg\min}\ J(\Omega, V_b) \\ J(\Omega, V_b) &= \sum_{m=1}^{M} |\gamma^m\, P_m(\Omega, V)|^2 \end{aligned} \quad (19)$$

Thus we are faced with a minimization problem with constraint $V_{b1} > 0$, and $\|V_b\| = V > 0$ which is the speed of the MAV. The idea is to make $|P_m(\Omega, V_b)|$ as small as possible weighted by the reliability of $(\dot{x}^m, \dot{y}^m)$.

Let us now consider the existence of solutions for this optimization problem. The matrix $A(x,y)$ in Equation (16) has eigenvalues:

$$0,\ \frac{1}{2}\left(1 + x^2 \pm \sqrt{(1+x^2)^2 - 4\,(1+x^2)y^2 - 4y^4}\right).$$

The right eigenvector corresponding to the 0 eigenvalue is $v_r(x,y) = [x\ \ y\ \ 1]^T$, while the left eigenvector is $v_l(x,y) = [-1\ \ -y\ \ x]$. It can also be easily checked that $v_l(x,y) \cdot B(x,y) = [0\ \ 0]$.

The physical meaning of $v_l^T(x,y)$ and $v_r(x,y)$ is as follows. Substituting these vectors for V_b and Ω in Equations (14-15) we get $\dot{x} = 0$ and $\dot{y} = 0$. Therefore at every point (x,y) there is an ambiguity in the estimation of the linear velocity in the direction $v_l^T(x,y)$ and the angular velocity in the direction $v_r(x,y)$. However, this ambiguity can be partially resolved by knowledge of $(\dot{x}^m, \dot{y}^m)$ at several points $m = 1, \cdots, M$ where $M \ge 5$. In the equation: $P(\Omega, V_b) = 0$, we still have the issue that if $V_b \neq 0$ is a solution, then any $\alpha\, V_b$ is a solution for $\alpha \neq 0$. This is resolved by the two constraints on the optimization problem, so that we have a unique solution.

We need the following non-singularity condition for the structure blocks.

Definition 3.1 (Non-singularity condition): The set of structure blocks together with the estimated motion vectors $\{((x^m, y^m), (\dot{x}^m, \dot{y}^m), \gamma^m);\ m = 1, \cdots, M\}$ where the reliability indices $\gamma^m > 0$, are said to form a *non-singular set* if for each $\xi \in \mathbb{R}^3$ the set of vectors:

$$\Upsilon \triangleq \left\{A(x^m, y^m)\,\xi + B(x^m, y^m)\begin{bmatrix}\dot{x}^m \\ \dot{y}^m\end{bmatrix};\right. \\ \left. m = 1, \cdots, M\right\}$$

contains at least one non-zero vector.

It is clear that this is a necessary condition for the solution of the problem, because otherwise $J(\Omega, V_b)$ would be zero for some spurious value of angular velocity.

Theorem 3.1: Assume that the axis of the camera is pointed along the nose of the aircraft, and that the distance of the external objects from the camera is much greater than the distance of the focus of the camera from the center of mass of the aircraft. Suppose that the true linear speed $\|V_{b,true}\|$ is known at some instant of time T_k. Suppose that the component of the inertial velocity along the Z axis of the camera in Figure 3 is positive, that is, $V_{b1,true} > 0$. Furthermore, suppose that the number of structure blocks $M \geq 5$, and that the set of structure blocks form a non-singular set for the reliability based motion analysis. Denote the true velocities of the air vehicle at instant T_k by $(\Omega_{true}, V_{b,true})$. Then, there exists a unique solution (Ω^*, V_b^*) to the optimization problem (19)at time T_k, and this solution coincides with the true solution $(\Omega_{true}, V_{b,true})$ if and only if (i) the speed V is known at time T_k; (ii) the vectors $(\dot{x}^m, \dot{y}^m)$ for the structure blocks (x^m, y^m) are estimated correctly and coincide with the true values $(\dot{x}^m_{true}, \dot{y}^m_{true})$ at time T_k.

Proof: The harder part of the claim is the *if* part that we prove first. Observe that the cost function in (19) is quadratic as a function of (Ω, V_b) and that $J(\Omega_{true}, V_{b,true}) = 0$ as $P(\Omega_{true}, V_{b,true}) = 0$. This means that in the absence of noise, the algorithm will converge to a point in the equivalence class $\{(\Omega, V_b) \mid J(\Omega, V_b) = 0\}$. We need to show that this equivalence class consists of only one point $(\Omega_{true}, V_{b,true})$.

The reason for $M \geq 5$ is that there are 5 parameters to be estimated (3 for Ω and 2 for the direction of the unit vector $\frac{V_b}{\|V_b\|}$)and so we need at least 5 equations for the structure blocks. Another preliminary observation is that the set of points $\{(\Omega, 0);\ \Omega \in \mathbb{R}^3\}$ lead to $P(\Omega, 0) = 0$. However, these points are eliminated by the constraint $\|V_b\| = V > 0$.

If $(\Omega_{true}, V_{b,true})$ is the true solution, and the result of the reliability-based motion estimation (see [1]) is error-free (that is, $(\dot{x}^m, \dot{y}^m);\ m = 1, \cdots, M$ exactly satisfies Equations (14 - 15)), then we will show that the result of the optimization is $(\Omega^*, V_b^*) = (\Omega_{true}, V_{b,true})$. By rewriting (16) we get:

$$P(\Omega, V_b) = \frac{1}{Z}\ V_b \cdot (v_l \times V_{b,true} + A(x,y)\,(\Omega - \Omega_{true})).$$

Clearly, if $(\Omega, V_b) = (\Omega_{true}, V_{b,true})$ then $P(\Omega, V_b) = 0$. Now suppose $P(\Omega, V_b) = 0$ for some fixed (Ω, V_b) values. If $\Omega \neq \Omega_{true}$ then the second term inside the parentheses in the equation above is non-zero for a generic point (x, y). It is also a quadratic function of (x, y) by the definition of $A(x, y)$. The first term inside the parentheses is a linear function of (x, y) for a given vector $V_{b,true}$. Hence for a generic point (x, y) the term inside the parentheses is not zero and is a quadratic function of (x, y). As V_b is a constant, $P(\Omega, V_b)$ cannot be zero for a generic point (x, y), which implies that our assumption of $\Omega \neq \Omega_{true}$ is false.

Next suppose that $\Omega = \Omega_{true}$. Then we have:

$$P(\Omega, V_b) = \frac{1}{Z}\ V_b\ \cdot\ v_l \times V_{b,true},$$

which is zero for any generic point (x, y) if and only if $V_b = \alpha\, V_{b,true}$, where $\alpha \in \mathbb{R}$. Due to the constraint $\|V_b\| = \|V_{b,true}\|$, we must have $V_b = \pm V_{b,true}$. Now the second constraint $V_{b,1} > 0$ combined with the given condition $V_{b,1true} > 0$ implies that $V_b = V_{b,true}$.

The easier *only if* part only requires the observation that if $(\Omega, V_b) = (\Omega_{true}, V_{b,true})$, then $(\dot{x}, \dot{y}) = (\dot{x}_{true}, \dot{y}_{true})$ by Equations (14-15). Furthermore, knowledge of $V_{b,true}$ implies the knowledge of the speed V. □

B. Range Estimation

Once the camera motion (Ω, V_b) is computed through either of the Methods I or II, we can determine the range (or depth) Z for each block in the scene. As discussed earlier and detailed in [1], the image is partitioned into blocks, $\mathbf{B}^n$, $1 \leq n \leq P$. If the $(\dot{x}^m, \dot{y}^m);\ \ m = 1, \cdots, M$ are the motion vectors computed for the structure block (x^n, y^n) using the reliability based estimation scheme [1], then the range Z^m for these blocks can be determined by least mean squared error estimation:

$$\begin{aligned} Z^m \quad &= \quad \arg\min_Z [\dot{x}^m - f(x^m, y^m, Z)]^2 \\ &\quad + [\dot{y}^m - g(x^m, y^m, Z)]^2; \end{aligned} \qquad (20)$$

where:

$$f = \frac{V_{b1}}{Z}(x + \frac{V_{b3}}{V_{b1}}) + \Omega_X xy - \Omega_Y(1 + x^2) + \Omega_Z y,$$
$$g = \frac{V_{b1}}{Z}(y - \frac{V_{b2}}{V_{b1}}) + \Omega_X(1 + y^2) - \Omega_Y xy - \Omega_Z x,$$

where we have suppressed the arguments on the LHS for brevity.

Let $\Lambda^n = \{(\dot{x}^n_j, \dot{y}^n_j) | 1 \leq j \leq L^n\}$ be the top candidate motion vectors for the n-th *non-structure motion block*. Recall that the non-structure blocks are not used in the computation of (Ω, V_b) and hence we may have multiple vectors for a non-structure block. If the block that corresponds to an object in the scene is stationary, the true motion vector must satisfy Eqs. (14-15). Observe that the functions $f(x, y, \cdot)$ and $g(x, y, \cdot)$ are affine functions of $\frac{1}{Z}$ for each x and y. For each candidate motion vector $(\dot{x}^n_j, \dot{y}^n_j)$ for the non-structure block (x^n, y^n), we can compute the corresponding range by orthogonal projection (see Figure 4):

$$Z^n_j = \arg\min_Z\ [\dot{x}^n_j - f(x^n, y^n, Z)]^2 + [\dot{y}^n_j - g(x^n, y^n, Z)]^2.$$

The corresponding fitting error is denoted by

$$E^n_j = [\dot{x}^n_j - f(x^n, y^n, Z^n_j)]^2 + [\dot{y}^n_j - g(x^n, y^n, Z^n_j)]^2.$$

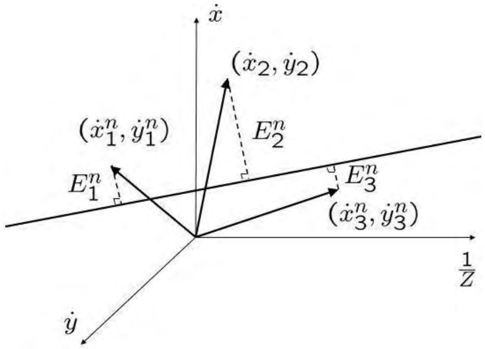

Fig. 4. Range Estimation for non-structure blocks

We choose the motion vector in the collection Λ^n to be the one with the least fitting error:

$$j^* = \underset{j=1,\cdots,L^n}{\arg\min}\; E_j^n. \tag{21}$$

The range of the block is given by Z^{j^*}, and the associated motion vector is $(\dot{x}_{j^*}^n, \dot{y}_{j^*}^n)$.

C. Orientation Estimation

One of the important uses of vision-based estimation is the possibility of computing the orientation of the vehicle with little additional information than that used in Method II. If at some instant of time $T_k;\ 0 \le k \le N$, one had accurate knowledge of the inertial velocity $V_{i,true}(T_k)$ through a GPS device; and in addition the conditions of Theorem 3.1 are satisfied so that $V_{b,true}(T_k)$ is known, then it is possible to compute the orientation $Q_{ib}(T_k)$. This computation does not rely on earlier estimates of $Q_{ib}(T_k)$. $V_{i,true}$ is related to $V_{b,true}$ according to:

$$V_{i,true} = Q_{ib}\, V_{b,true}. \tag{22}$$

Consider $V_{i,true}$ and $V_{b,true}$ as vectors in the same coordinate system. Then by Euler's theorem [13], there exists a unit vector ω such that:

$$V_{i,true} = \mathrm{Exp}(\hat{\omega}\,\theta)\, V_{b,true},$$

where Exp denotes the matrix exponential, and $\hat{\omega}$ denotes the skew-symmetric matrix derived from ω that satisfies $\hat{\omega}\, r = \omega \times r$ for any vector r. The angle θ is the angle of rotation from $V_{b,true}$ to $V_{i,true}$. The direction of rotation ω is perpendicular to both $V_{b,true}$ and $V_{i,true}$, and hence:

$$\omega = \frac{V_{b,true} \times V_{i,true}}{\|V_{b,true} \times V_{i,true}\|}.$$

Now, $\|V_{b,true}\| = \|V_{i,true}\| = V > 0$ is the speed of the MAV, and we have the equations:

$$V_{b,true} \cdot V_{i,true} = V^2 \cos\theta,$$
$$\|V_{b,true} \times V_{i,true}\| = V^2 \sin\theta.$$

These two equations can be used to compute θ without ambiguity, and we get $Q_{ib} = \mathrm{Exp}(\omega)$.

IV. Conclusion

In this paper, we have considered the problem of velocity and range estimation for a UAV using a camera and the knowledge of the linear speed of the UAV. Together with [1], we have shown that the ego-motion problem can be solved by using a reliability-based motion computation, followed the solution of a well-posed constrained optimization problem. Theorem 3.1 complements the well known classical eight-point algorithm found in the literature and is numerically simpler to implement. Once the velocities have been found, the range of the objects can be computed easily.

V. Acknowledgement

R. Iyer and Z. He were supported by ASEE/AFOSR Summer Faculty Fellowships during the summer of 2005. Discussions with Prof. M. Pachter of AFIT, and Prof. D. Chichka of GWU are acknowledged.

References

[1] Z. He, R. V. Iyer and P. R. Chandler Vision-based UAV flight control and obstacle avoidance. *to appear in IEEE Automatic Control Conference 2006, Minnesota, MN.*

[2] M. Polat and M. Pachter. INS aiding by tracking an unknown ground object: - theory. In *Proceedings of the IEEE Automatic Control Conference*, June 4-6 2003. Denver, CO.

[3] P. Merrell, D. Lee, and R. Beard. Statistical analysis of multiple optical flow values for estimation of unmanned aerial vehicle height above ground. In D. P. Casasent, E. L. Hall, and J. R editors, *Proceedings of SPIE: Intelligent Robots and Computer Vision XXII: Algorithms, Techniques and Active Vision*.

[4] P. Merrell, D. Lee, and R. Beard. Obstacle avoidance for unmanned air vehicles using optical flow probability distributions. In D. W. Gage, editor, *Proceedings of SPIE: Mobile Robots XVII*, volume 5609, pages 13 – 22, Bellingham, WA, 2004.

[5] Y. Ma, S. Soatto, J. Košecká, and S. S. Sastry. *An Invitation to 3-D Vision* . Springer Verlag, 2004

[6] R. V. Iyer, R. W. Holsapple, and P. R. Chandler. A new method for the computation of motion from image sequences. In *Proceedings of the IEEE Automatic Control Conference*, June 2005. Portland, OR.

[7] Oliver Faugeras. *Three-Dimensional Computer Vision - A Geometric Viewpoint*. MIT Press, Cambridge, Massachusetts, 1993.

[8] E. P. Simoncelli, E. H. Adelson, and D. J. Heeger. Probability distributions of optical flow. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 310–315, Maui, HI, June 1991.

[9] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[10] Bernard Etkin. *Dynamics of Atmospheric Flight*. John Wiley & Sons, New York, 1972.

[11] J.K. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images - a review. In *Proceedings of the IEEE*, volume 76, pages 917–935, August 1988.

[12] Z. Duric and A. Rosenfeld. Shooting a smooth video with a shaky camera. *Machine Vision and Applications*, 13:303–313, 2003.

[13] E. T. Whittaker. *A treatise on the analytical dynamics of particles and rigid bodies*. Cambridge Mathematical Library, fourth edition, 1993.

[14] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:580–593, 1997.