

Radar Odometry on Small Unmanned Aircraft

Eric B. Quist* and Randal W. Beard†

Brigham Young University, Provo, UT, 84602, USA

This paper develops an algorithm to estimate motion using a radar and ground targets. It involves estimating motion using an Extended Kalman Filter (EKF) with an Inertial Measurement Unit (IMU) and a side-looking Synthetic Aperture Radar (SAR) carried on a fixed wing aircraft flying over unknown, flat terrain. The accuracy of the motion estimation is compared to dead reckoning using only the IMU, with truth data being provided by a standard IMU/GPS Kalman filter. Initial results show that over 4.5km of simulated flight, position drift of around 300m resulted, as compared to 2.5km using only the IMU.

I. Introduction

True Autonomous Navigation by an Unmanned Air Vehicle (UAV) is predicated on the UAVs ability to recognize its position relative to the surrounding environment. Current navigation systems typically use an Inertial Measurement Unit (IMU) in conjunction with a GPS sensor. In these systems, a Kalman Filter (or Extended Kalman Filter) uses the IMU to propagate the vehicles position and GPS to correct the drift introduced by the IMU. Such systems provide accurate position measurement, but are reliant on the reception of GPS positioning. As GPS signals are easily jammed, solutions that don't rely on GPS are necessary for true autonomous navigation.

Several approaches exist for GPS-denied navigation. One option involves using interoceptive sensors such as an IMU to estimate the relative motion of the aircraft. IMUs can be quite accurate, and are often used successfully on inter-ballistic missiles, but that accuracy requires a very heavy and expensive sensor that is often inappropriate for small, unmanned systems. Smaller, cheaper IMUs, such as those typically found on small UAVs, have a large amount of drift, thus limiting the needed accuracy to small windows of time.

To reduce their dependence on IMUs, many algorithms, such as Nister's Visual Odometry (VO) algorithm,¹⁰ estimate relative motion using the change in perspective of consecutive camera images. While VO algorithms have been shown to be accurate and can operate real-time,^{4,11} they are severely constrained by the range limitations of optical sensors as well as by their dependence on good weather and daytime navigation (or the use of lighting).

Radar, as compared to vision, has better range resolution and is not limited by environmental factors such as time-of-day, fog, or rain. The use of radar for motion estimation from a moving platform has been investigated,¹³ but the Size, Weight, and Power (SWaP) requirements of these systems, as well as their cost, has limited the scope and availability of such systems.

Over the last decade, recent technological advancements have resulted in significant decreases in SWaP for many radar systems, such as Synthetic Aperture Radar (SAR).^{1,5,12} Miniaturization allows radar to be an optional payload on small UAVs, and thus considered as an optional sensor to provide motion estimation. Using radar, as opposed to vision, will provide precise range measurements to ground reflectors, while also allowing for night-time and all-weather operation. While some related results have been reported,⁶⁻⁹ there is currently no working Radar Odometry system.

The contribution of this paper is to describe a Radar Odometry system that limits the drift rate of an IMU-based navigation systems. At a later stage we envision using this system in conjunction with a radar-based placement recognition solution to provide a complete GPS-denied navigation solution.

*Graduate Research Assistant, Electrical and Computer Engineering 459 CB; quist@byu.edu.

†Professor, Electrical and Computer Engineering 459 CB; beard@ee.byu.edu; AIAA Senior Member.

The Radar Odometry motion estimation approach will be described in Section II. Section III describes the Kalman filter used to estimate motion using an IMU, and an IMU in conjunction with Radar Odometry. In Section IV, the simulation results of each motion estimation approach will be discussed.

II. Radar Odometry

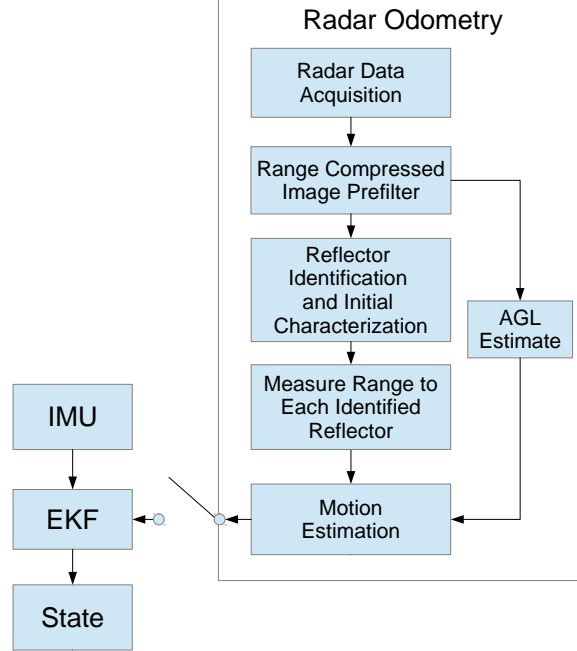


Figure 1. Block Diagram describing the aircraft state estimation using an IMU and the optional Radar Odometry algorithm.

The Radar Odometry algorithm is outlined in Figure 1. The algorithm performs an initial range compressed image pre-filter. The filtered image is then used to identify and characterize reflectors with large radar cross section. The range to each identified reflector is then measured. The height above ground level (AGL) is also estimated using the range compressed image. The aircraft's motion is measured using the range measurements in conjunction with the AGL estimate.

II.A. Range Compressed Images & Pre-Filtering

There are many approaches to transmitting and receiving radar data. In this paper, we focus on using a Linear Frequency Modulated Continuous Wave (LFM-CW) Radar, which returns the range to reflectors observed in the beamwidth of the radar's single aperture (one antenna).

II.A.1. Radar Range and LFM-CW Radar

LFM-CW radar involves repeatedly performing radar transmits and receives, or chirps. The frequency of chirp repetition is referred to as the pulse repetition frequency (PRF),

$$PRF = \frac{1}{\tau},$$

where τ indicates a chirp duration. During a chirp, the LFM-CW radar transmits a single, linear frequency modulated chirp, that starts at time t , and is given by

$$x_t(t) = a_t(t) \cos(2\pi F(t)t),$$

where

$$a_t(t) = u(t) - u(t - \tau)$$

represents the transmit pulse, $u(t)$ is the unit step function, and where

$$F(t) = F_0 + \frac{\beta}{\tau}t$$

indicates the frequency as a function the initial transmit frequency F_0 , transmit bandwidth β , and time t .

As the transmitted signal is reflected, there is a delay in time, Δt_i , unique to each reflector, i . This time delay represents the delay to the reflector and back to the receiver, resulting in the delay

$$\Delta t_i = 2 \frac{r_i}{c},$$

where r_i is the range to reflector i and c is the speed of light. After a reflector reflects the transmitted signal, the reflected signal received by the radar is

$$x_r(t) = a_r(t) \sigma_{reflector} x_t(t - \Delta t_i),$$

where

$$a_r(t) = u(t) - u(t - \tau)$$

is the receive window, $\sigma_{reflector}$ is the reflector's radar cross section.

The transmit and receive signals are mixed, resulting in

$$\begin{aligned} x_r(t) \otimes x_t(t) &= a_t(t) \cos(2\pi F(t)t) \cdot \\ &\quad a_r(t) \sigma_{reflector} x_t(t - \Delta t_i) \\ &= a_t(t) a_t(t - \Delta t_i) a_r(t) \sigma_{reflector} \cdot \\ &\quad \cos(2\pi F(t)t) \cdot \\ &\quad \cos(2\pi F(t - \Delta t_i)(t - \Delta t_i)). \end{aligned}$$

Range compressing a chirp involves performing the Fourier Transform on the mixed transmit and receive signals:

$$X_m(jw) = \mathcal{F}(x_r(t) \otimes x_t(t)).$$

As the reflector radar cross-section is unknown and range dependent, the mixed transmit and receive is approximated as

$$\begin{aligned} x_r(t) \otimes x_t(t) &\approx \cos(2\pi F(t)t) * \\ &\quad \cos(2\pi F(t - \Delta t_i)(t - \Delta t_i)), \end{aligned}$$

resulting in the approximate range compressed signal

$$\begin{aligned} X_m(jw) &\approx \left(\frac{\sin\left(\pi\left(\omega - \frac{\beta}{\tau}\Delta t_i\right)\tau\right)}{\pi\left(\omega - \frac{\beta}{\tau}\Delta t_i\right)\tau} \right)^2 \\ &= \text{sync}\left(\omega - \frac{\beta}{\tau}\Delta t_i\right), \end{aligned}$$

which is the sync function centered at the range-dependent frequency

$$\omega = \frac{\beta}{\tau}\Delta t_i = \frac{\beta}{\tau} \frac{2r_i}{c},$$

where r_i is the range to reflector i and c is the speed of light.

The range compressed chirp represents the accumulative strength of all radar return for a given range r during the specified chip. The mapping from range bin index, b , to range is defined as

$$r = r_0 + br_{res}, \tag{1}$$

where r_0 is the minimum range bin visible to the radar, and r_{res} is the radar's range resolution.

Therefore the radar measurement over the time window $[t - \tau, t]$ of the chirp can be thought of as a column vector where each row index (range bin) represents a particular range, and the value at that index represents the strength of radar return. Since chirps occur sequentially in time, we define the chirp index

$$s = \frac{t - t_0}{\tau}$$

to be the chirp that occurs over the time window $[t_0 + (s - 1)\tau, t_0 + s\tau]$, where t_0 corresponds to the absolute time the first chirp in the image was started.

Stacking the range measurements at consecutive chirps results in a positive matrix I_{RC} called the range compressed image as it can be displayed and visualized as an image. The pixel $I_{RC}[b, s]$ is the strength of the radar return from chirp s at range bin b . Figure 2 shows a simulated (i.e., ideal) range compressed image as the radar moves past two reflectors while traveling a straight line at constant velocity. Prior to chirp 600, the antenna beamwidth renders both reflectors unobservable. Once observable, the range to the reflectors decreases until chirps 2000 and 2300, which occurs as the aircraft flies past each reflector. The range to the reflectors then increases until chirp 3450, at which time the antenna beamwidth renders both reflector unobservable. A more thorough treatment of the range compression derivation may be found in.³

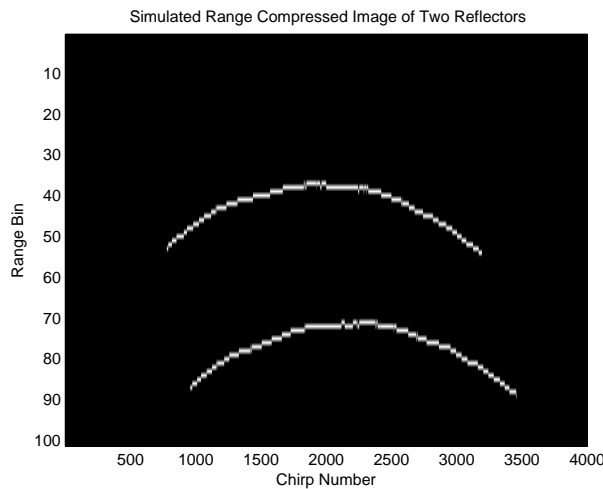


Figure 2. Range compressed image of two reflectors, with the chirp number corresponding to specific transmit/received pair and the range bin corresponding to the range of the observed reflector.

II.A.2. Range-Compressed Image Pre-Filtering

A range compressed image, generated by a LFM-CW Synthetic Aperture Radar, contains a significant amount of noise speckle, as may be demonstrated in Figures 3a and 3b, which shows a range compressed image from a flight and simulation respectively. While the reflectors, as seen by the hyperbolic lines, indicate the relative motion of the aircraft, the noise in the imagery limits the ability to distinguish individual reflectors.

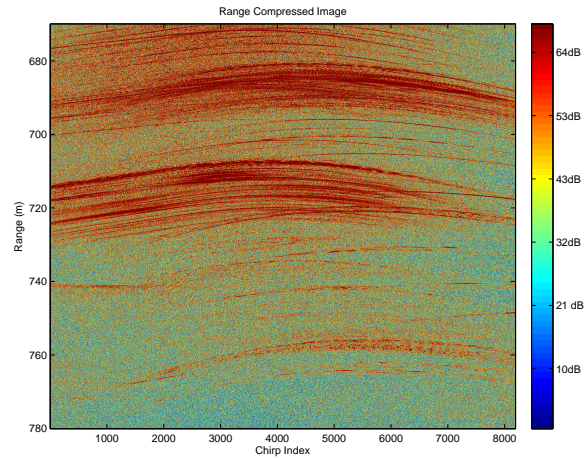
To remove the noise, several techniques are implemented. Rather than removing the average pixel value from the entire image, a weighted average pixel value for a each pixel's 9x13 neighborhood is removed from the image. Additionally, as the ranges to each reflector changes very little in comparison to the chirp index, a weighted horizontal corner kernel,

$$k_{hc} = \begin{bmatrix} -1 & 2 & -1 \end{bmatrix},$$

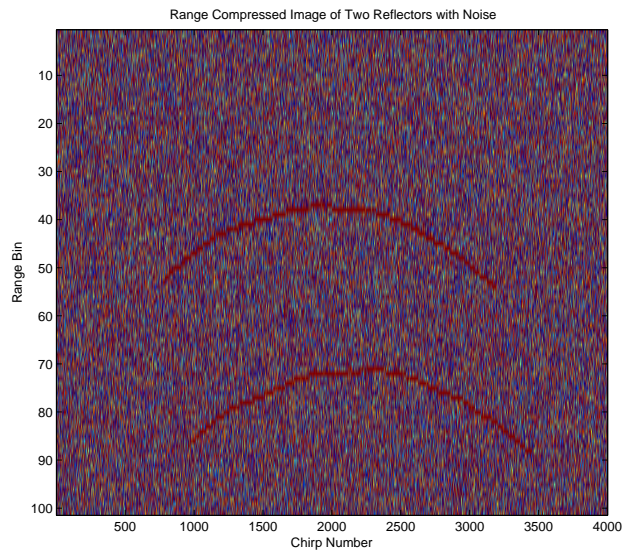
is also removed from the range compressed image. The weighted image is then thresholded, resulting in the filtered image I_F shown in the prefiltered flight image in Figure 4a, while 4b shows the prefiltered simulated image.

II.B. Reflector Identification and Initial Characterization

While pre-filtering the range compressed image removes much of the noise, it is still necessary to identify individual reflectors and estimate the aircraft's range to each reflector. Initially, this is performed by identi-



a)



b)

Figure 3. Range compressed image of radar return during a)an aircraft flight and b)simulation.

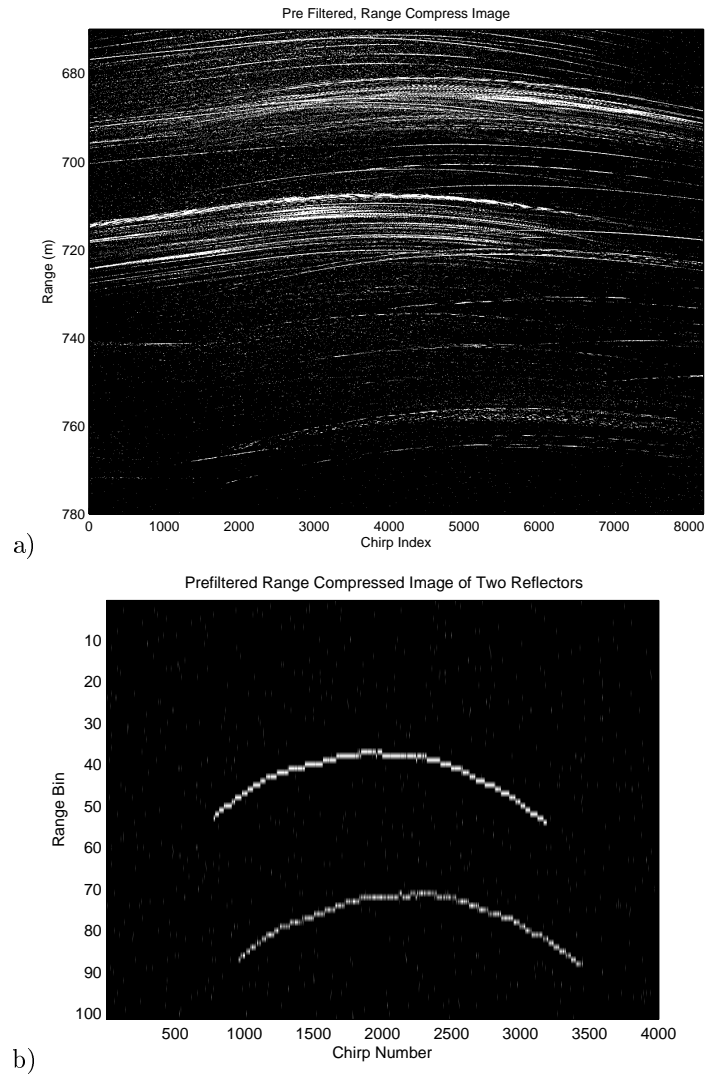


Figure 4. Range compressed image of radar return during an simulated flight a) prior to filtering and b) pre-filtered.

flying individual reflectors and performing a first-order estimate of the motion of the aircraft relative to each reflector.

II.B.1. Range to Reflector during flight

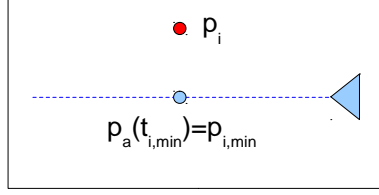


Figure 5. Possible Aircraft Flight Track

Consider Figure 5, which shows an aircraft, represented by a triangle, as it flies in a straight line past reflector i , represented by the red circle, located at \mathbf{p}_i . The position of the aircraft $\mathbf{p}_a(t)$, when it is closest to \mathbf{p}_i , as identified by the blue circle, is defined as $\mathbf{p}_{i,min}$ and occurs at time $t_{i,min}$. Defining the aircraft position as a function of $t_{i,min}$ results in

$$\mathbf{p}_a(t) = \mathbf{p}_{i,min} + \dot{\mathbf{p}}_a(t)(t - t_{i,min}).$$

The range to reflector i is represented by $r_i(t)$, while the squared range is calculated as

$$\begin{aligned} r_i(t)^2 &= (\mathbf{p}_a(t) - \mathbf{p}_i)^T (\mathbf{p}_a(t) - \mathbf{p}_i) \\ &= (\mathbf{p}_{i,min} - \mathbf{p}_i)^2 \\ &\quad - 2\dot{\mathbf{p}}_a(t)(t - t_{i,min})(\mathbf{p}_{i,min} - \mathbf{p}_i) \\ &\quad + \dot{\mathbf{p}}_a(t)^2(t - t_{i,min})^2. \end{aligned}$$

The straight flight and fixed velocity assumptions imply that $\dot{\mathbf{p}}_a(t)$ and $\mathbf{p}_{i,min} - \mathbf{p}_i$ are terms becoming orthogonal, resulting in

$$r_i(t)^2 = (\mathbf{p}_{i,min} - \mathbf{p}_i)^2 + \|\dot{\mathbf{p}}_a\|^2(t - t_{i,min})^2.$$

Further defining the minimum range to reflector i as $r_{i,min}$ and the aircraft speed $V_g = \|\dot{\mathbf{p}}_a\|$, results in a hyperbolic equation for the range equation:

$$\frac{1}{r_{i,min}^2} r_i^2(t) - \frac{V_g^2}{r_{i,min}^2} (t - t_{i,min})^2 = 1. \quad (2)$$

To express this equation in range compressed image coordinates, let

$$r_i[s] = r_0 + b_i[s] r_{res},$$

which is a function of the range bin $b_i[s]$ during chirp s . The corresponding minimum range equation is

$$r_{i,min} = r_0 + b_{i,min} r_{res}.$$

Similarly define $t_{i,min}$ and $s_{i,min}$ so that

$$t_{i,min} = s_{i,min} \tau - t_0.$$

The hyperbolic range equation using the discrete range bin and chirp number becomes

$$\frac{(r_0 + b_i[s] r_{res})^2}{(r_0 + b_{i,min} r_{res})^2} - \frac{V_g^2 (s - s_{i,min})^2 \tau^2}{(r_0 + b_{i,min} r_{res})^2} = 1. \quad (3)$$

II.B.2. Reflector Detection Using the Hough Transform

As r_0 and r_{res} are constant, and predefined, each reflector is uniquely defined by V_g , $b_{i,min}$, and $s_{i,min}$ (see Equation 3) which constrain the hyperbolic shape of the reflector in the range compressed image. A hyperbolic Hough Transform² is used to identify, and provide initial parametrization for each reflector. This is performed by creating a parameter space

$$H = \mathbb{R} \times \mathbb{N} \times \mathbb{N},$$

containing sufficient hyperbola parameter combinations,

$$H \in (V_g, b_{i,min}, s_{i,min}),$$

to represent the observable hyperbolas in the range compressed image. Specifically, the Hough Transform iterates over pixels in the pre-filtered image. When a pixel is illuminated, it votes for all possible parameter combinations that would result in the specific pixel being illuminated. Combinations of parameters that received large numbers of votes suggest that a hyperbola is present with those parameters.

Ideally, when all illuminated pixels have been traversed, the parameter combinations that receive the largest number of votes would be used to parametrize and identify reflector hyperbolas found in the image. However, non-straight flight, non-constant airspeed, measurement inaccuracies, multiple reflectors, and the resolution of hyperbola constraints often result in a large number of votes being cast for incorrect constraints. Using the Hough Transform to detect reflectors from the prefiltered image (seen in 4b) results in Figure 6, which shows an image representation of the Hough space

$$I_{V_g}(b_{i,min}, s_{i,min}) = H(\hat{V}_g, b_{i,min}, s_{i,min}) \quad (4)$$

for a single, fixed \hat{V}_g . As the Hough Transform is a voting algorithm, neighboring range bins and chirp numbers often have similar numbers of votes. Additionally, possible $b_{i,min}$ and $s_{i,min}$ combinations that could result in pixels being illuminated in multiple hyperbolas (as seen in Figure 6b near $s_{i,min}$ 1500 and $b_{i,min}$ 50) are also artificially high. To this end, additional filtering is necessary to select the reflectors, and their approximate hyperbola parameters.

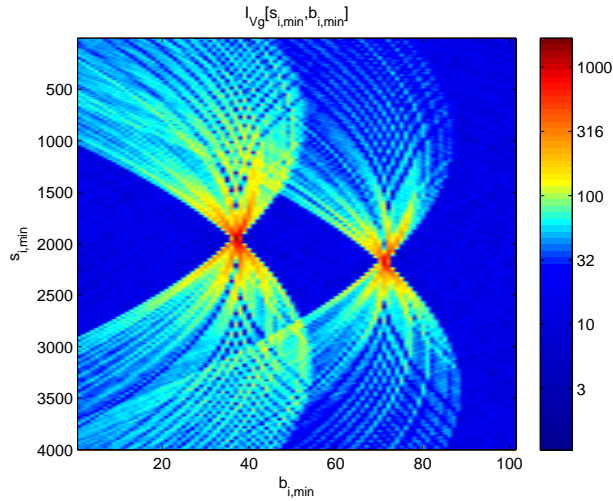


Figure 6. The Hough-space image I_{V_g} for a fixed \hat{V}_g , with color indicating the number of votes.

To further isolate individual hyperbola constraints, a ground velocity \hat{V}_g is estimated to be the ground velocity in the Hough-space cube that contains the pixel with the largest number of votes. The resulting two dimensional Hough-space image (see Equation 4) is then used to identify specific reflectors. This is done by removing the average pixel value, $E[I_{\hat{V}_g}]$ from the image, resulting in

$$I_{mr} = I_{\hat{V}_g} - E[I_{\hat{V}_g}].$$

The mean-removed image I_{mr} is normalized, such that

$$I_n = \frac{I_{mr}}{\max_{b,s}(I_{mr})},$$

then thresholded, resulting in $I_m = I_n > T_n$, where T_n is the threshold applied. The pixels illuminated in the thresholded image are then segmented into connected groups G_{V_g} . Figure 7a shows the thresholded Image generated from the Hough-space image in Figure 6b. As each of the two sets of unconnected pixels exist, two groups are formed.

Erroneous peaks with-in each group are removed from I_{V_g} by convolving the image using a 5x5 smoothing kernel, resulting in the smoothed image I_{sm} . The pixel with the maximum value in I_{sm} from each group $g \in G_{V_g}$ is identified as a reflector and characterized by Hough-space indexing parameters \hat{V}_g , $\hat{b}_{i,min}$, and $\hat{s}_{i,min}$. To limit the number of hyperbolas, only groups containing more than T_g pixels are considered, where T_g identifies a minimum group count threshold. Figure 7b shows the smoothed image resulting from 6b. The pixel with the largest number of votes from each group (see 6a) are identified as reflectors and characterized by their $\hat{s}_{i,min}$ and $\hat{b}_{i,min}$ indices.

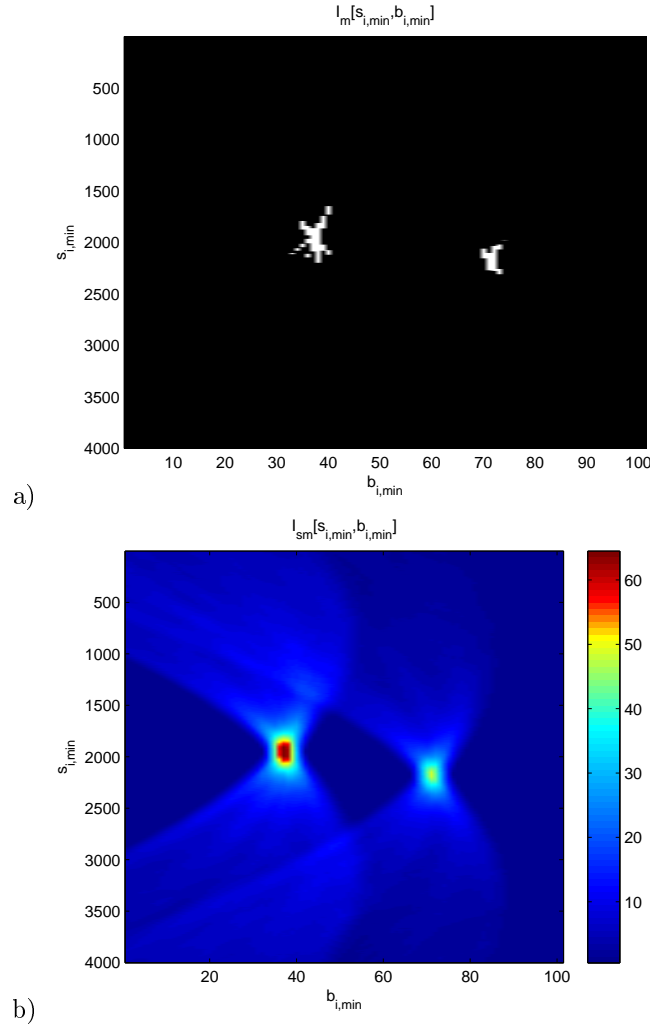


Figure 7. (a) Thresholded Hough-space image I_m , and (b) the smoothed Hough-space image I_{sm} , with color indicates the number of votes.

II.C. Range estimation

The initial reflector identification and parametrization calculated by the Hough Transform provides a rough estimate as to where each reflector is at each chirp. While imprecise, this estimate provides a starting point to determine a more precise range estimate for each chirp.

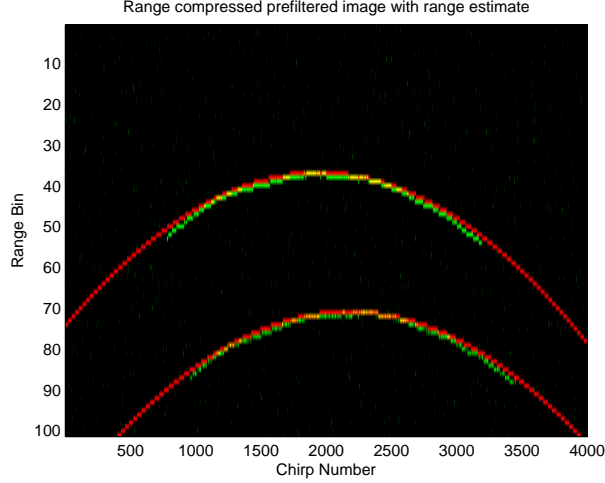


Figure 8. Range compressed image of an observed single reflector (seen in green) as compared to the initial hyperbola estimate (seen in red)

Consider Figure 8 which shows a pre-filtered radar's range compressed image with the observed range to the reflector seen in green. To compare the observed radar return with the initial hyperbola estimate, the estimated \hat{V}_g , $\hat{b}_{i,min}$, and $\hat{s}_{i,min}$ parameters are used to generate an initial hyperbolic range estimate at each chirp. For visualization, this hyperbola estimate is super-imposed in red on the radar's range compressed image and shown in Figure 8.

Estimating the range to the reflector during each chirp involves traversing the pre-filtered image (previously defined as I_F) and discerning between an observed reflector and noise. At each incremental chirp, s_{inc} , the range is estimated, resulting in $b_i[s_{inc}]$, using the range compressed image and an adapting range estimate, $\hat{b}_i[s_{inc}]$, which is initialized to the initial hyperbola formed by the estimated Hough parameters. Consider Figure 9a which shows a synthetic observed reflector, in green, and the initial range estimate, in red. Starting at the vertex of the hyperbola (seen in the left-most white square in Figure 9b), each increasing range bin is assessed. If the pixel at the estimated pixel location $I_F(\hat{b}[s_{inc}], s_{inc})$ is illuminated (or both $I_F(\hat{b}[s_{inc}] + 1, s_{inc})$ and $I_F(\hat{b}[s_{inc}] - 1, s_{inc})$ are illuminated), the range to target is estimated to be the adapted range estimate. This occurs for the first 3 assessed chirps, resulting in the results shown in Figure 9b, in white. When $I_F(\hat{b}[s_{inc}], s_{inc})$ is not illuminated (as seen by the red and green during chirp number 8 in Figure 9b), but $I_F(\hat{b}_i[s_{inc}] + 1, s_{inc})$ or $I_F(\hat{b}_i[s_{inc}] - 1, s_{inc})$ are illuminated, each adapting estimate $\hat{b}[s]$ value for all future chirps ($s \geq s_{inc}$) adjusted such that $I_F(\hat{b}[s_{inc}], s_{inc})$ becomes illuminated. When this occurs, the range estimate $b_i[s_{inc}]$ is set to the new $\hat{b}_i[s_{inc}]$. This is demonstrated by the resulting range compressed map shown in Figure 9c, which shows a shifted $\hat{b}_i[s]$ for $s \geq s_{inc}$. It also shows that the first adjusted range becomes the range estimate as shown in white.

The process continues until $I_F(\hat{b}_i[s_{inc}], s_{inc})$, $I_F(\hat{b}_i[s_{inc}] + 1, s_{inc})$, and $I_F(\hat{b}_i[s_{inc}] - 1, s_{inc})$ are all not illuminated (as seen in range bin 13 in Figure 9d). When this happens, the range estimate is assigned to the adapting estimate and an uncertainty estimate is incremented, as seen in the blue during chirp 13 in Figure 9e. Once the edge of the range compressed image is reached, the algorithm is repeated for decreasing chirp values, again starting at the vertex. The resulting range estimates may be seen in Figure 9f, where white represents range bins where the estimate was observed, while blue indicates an unobserved range bin. Reflectors are considered unobservable when they have an accumulated uncertainty above T_u .

II.D. AGL Estimation

Nadir (the return from the ground immediately below the UAV) is used to estimate the AGL. This is calculated as the first range bin in the range compressed image with a signal larger than the AGL threshold T_{AGL} , and is identified as d_{AGL} . The threshold is present to remove the measurement noise.

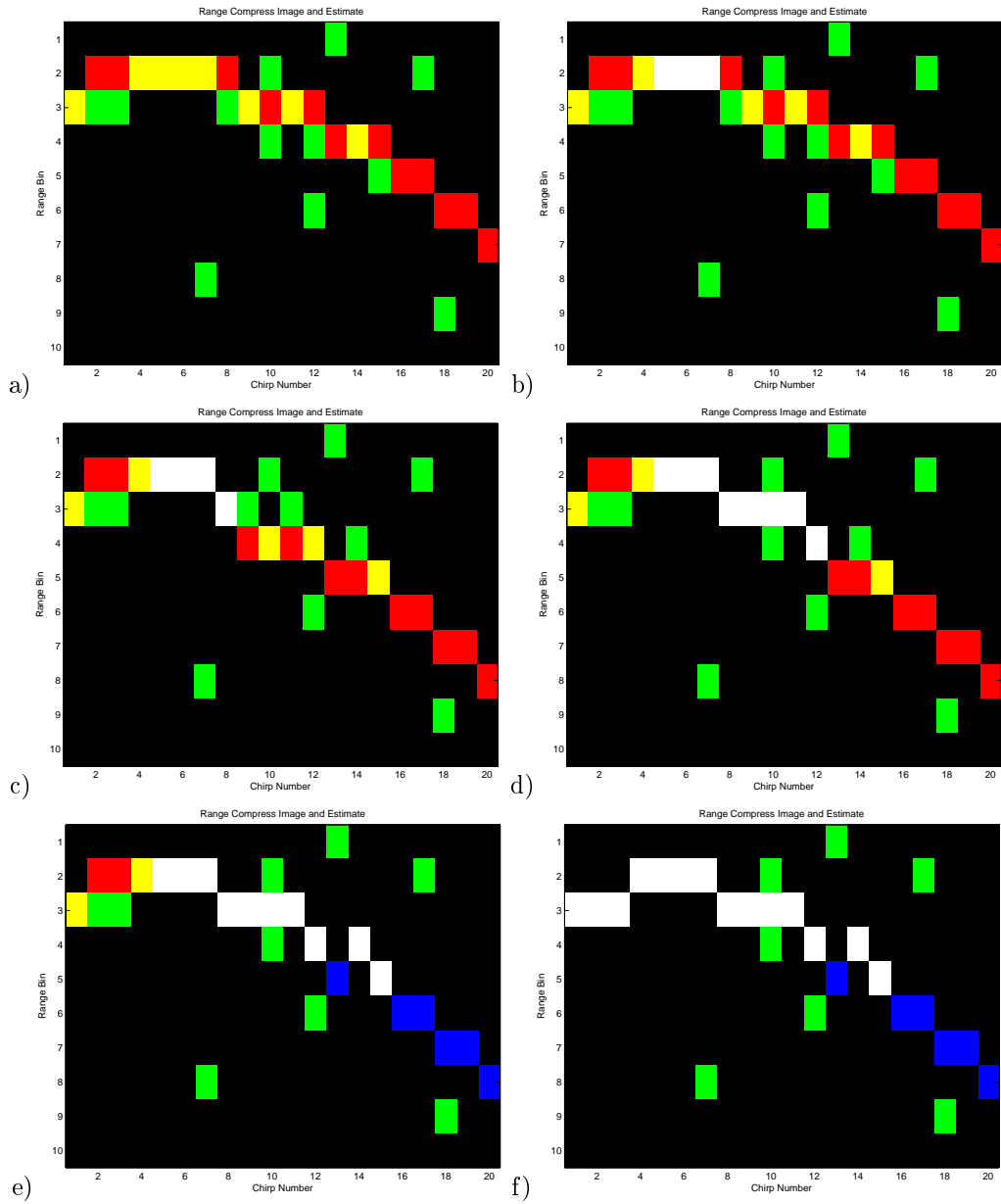


Figure 9. Synthetic range compressed image of an observed single reflector (seen in green) as compared to the initial hyperbola estimate (seen in red). Estimates that are observed are indicated in white, while uncertain estimates (ie. no measurement is considered an estimate) are seen in blue.

II.E. Motion Estimation

Using the estimated range to multiple reflectors, the motion of the aircraft is estimated. This is performed by calculating the incremental motion using the estimated range to consecutive pairs of reflectors. The ranges to each reflector, i and j , are both observed at $t_{i,min}$ and $t_{j,min}$. For simplicity, a northern flight-track is assumed. A north-east-down coordinate system is used, with down being relative to ground-level. A flat earth model used.

Define the position of the aircraft at $t_{i,min}$ as

$$\mathbf{p}_a(t_{i,min}) = \begin{bmatrix} n_{i,min} \\ e_{i,min} \\ d_{i,min} \end{bmatrix}.$$

At time $t_{j,min}$, the position of the aircraft may be represented as

$$\mathbf{p}_a(t_{j,min}) = \begin{bmatrix} n_{ij} + n_{i,min} \\ e_{ij} + e_{i,min} \\ d_{j,min} \end{bmatrix},$$

where n_{ij} and e_{ij} indicate the northern and eastern aircraft motion between $t_{i,min}$ and $t_{j,min}$. Figure 10 visualizes the aircraft at times $t_{i,min}$ and $t_{j,min}$. It also shows reflector i , located at

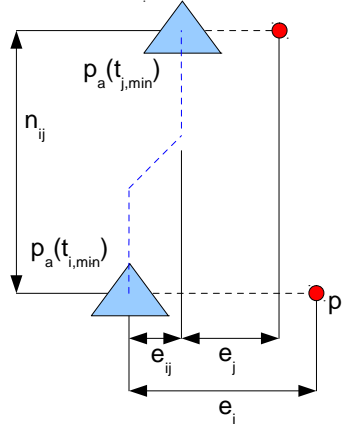


Figure 10. Motion experienced by an aircraft between times $t_{i,min}$ and $t_{j,min}$

$$\mathbf{p}_i = \begin{bmatrix} n_{i,min} \\ e_i + e_{i,min} \\ 0 \end{bmatrix},$$

and reflector j , located at

$$\mathbf{p}_j = \begin{bmatrix} n_{ij} + n_{i,min} \\ e_j + e_{ij} + e_{i,min} \\ 0 \end{bmatrix},$$

where e_i and e_j represent the eastern range to each reflector at \mathbf{p}_i and \mathbf{p}_j respectively.

Assuming that both reflectors are visible at times $t_{i,min}$ and $t_{j,min}$, the squared range to the reflectors are

$$\begin{aligned} r_i^2(t_{i,min}) &= e_i^2 + d_{i,min}^2 \\ r_j^2(t_{i,min}) &= n_{ij}^2 + (e_j + e_{ij})^2 + d_{i,min}^2 \\ r_i^2(t_{j,min}) &= n_{ij}^2 + (e_i - e_{ij})^2 + d_{j,min}^2 \\ r_j^2(t_{j,min}) &= e_j^2 + d_{j,min}^2. \end{aligned}$$

The down position of the aircraft is provided by the AGL measurement,

$$\begin{aligned} d_{i,min} &= d_{AGL}(t_{i,min}) \\ d_{j,min} &= d_{AGL}(t_{j,min}) \end{aligned}$$

which allows for the eastern position term for each target to be calculated as

$$\begin{aligned} e_i &= \sqrt{r_i^2(t_{i,min}) - d_{AGL}(t_{i,min})^2} \\ e_j &= \sqrt{r_j^2(t_{j,min}) - d_{AGL}(t_{j,min})^2}. \end{aligned}$$

Subtracting the cross-terms results in

$$r_j^2(t_{i,min}) - r_i^2(t_{j,min}) = e_j^2 - d_{j,min}^2 - e_i^2 + d_{i,min}^2 + 2e_{ij}(e_j + e_i).$$

Solving for e_{ij} gives

$$e_{ij} = \frac{r_j^2(t_{i,min}) - r_i^2(t_{j,min}) - e_j^2 + d_{j,min}^2 + e_i^2 - d_{i,min}^2}{2(e_j + e_i)}.$$

The northern motion may also be as solved as

$$n_{ij} = \sqrt{r_i(t_{j,min})^2 - (e_i - e_{ij})^2 - d_{j,min}^2},$$

which allows for the aircraft motion to be calculated using the measured ranges $r_i(t_{i,min})$, $r_j(t_{j,min})$, $r_i(t_{j,min})$, and $r_j(t_{i,min})$ in addition to the measured AGL $d_{AGL}(t_{i,min})$ and $d_{AGL}(t_{j,min})$.

Selecting reflector pairs p_i and p_j such that the reflectors are both visible at $t_{i,min}$ and $t_{j,min}$ involves sorting the reflectors by their respective t_{min} . Sequential reflectors are then selected and the resulting northern and eastern motion is calculated using each reflector pair.

III. Extended Kalman Filter

The radar motion estimation algorithm does not take advantage of other available sensors. Using IMUs in conjunction with the radar provides additional accuracy. Combining the multiple sensors with different update rates is often performed by a Kalman Filter, or an Extended Kalman Filter (EKF) when the system has non-linear dynamics.

III.A. The Extended Kalman Filter

The EKF, or Extended Kalman Filter, involves linearizing around the estimate and covariance of the current state. It involves predicting the state and using state observations to compensate for prediction error.

III.A.1. Prediction Model

The EKF prediction model is calculated

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1}),$$

resulting in a predicted covariance,

$$P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + G_kQ_uG_k^T + Q_{k-1}.$$

The calculation of F_{k-1} involves linearizing around $\hat{x}_{k-1|k-1}$ and u_{k-1} ,

$$F_{k-1} = \frac{\partial f}{\partial x}|_{\hat{x}_{k-1|k-1}, u_{k-1}},$$

while the calculation of G_k is also linearizing around $\hat{x}_{k-1|k-1}$ and u_{k-1} ,

$$G_k = \frac{\partial f}{\partial u}|_{\hat{x}_{k-1|k-1}, u_{k-1}},$$

III.A.2. Observation Model

The nonlinear observation estimate is defined as

$$\hat{y}_{k|k-1} = h(\hat{x}_{k|k-1}, u_k),$$

with covariance

$$S_k = H_k P_{k|k-1} H_k^T + R_k,$$

where H_k is linearized around $\hat{x}_{k-1|k-1}$ and u_{k-1} ,

$$H_k = \frac{\partial h}{\partial x} \big|_{\hat{x}_{k-1|k-1}, u_{k-1}}.$$

The observation error is defined as

$$\tilde{y}_k = y_k - h(\hat{x}_{k|k-1}, u_k),$$

while the predicted state is

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k.$$

The state covariance is calculated

$$P_{k|k} = (I - K_k H_k) P_{k|k-1},$$

where the Kalman Gain is defined

$$K_k = P_{k|k-1} H_k^T S_k^{-1}.$$

III.B. Sensor Models

III.B.1. IMU

An IMU sensor consists of both accelerometers and rate gyros. Each of the three accelerometers measure the acceleration along its axis, with each accelerometer typically aligned with one of the body-frame axis, resulting in

$$\begin{aligned} y_{accel,x} &= a_x + \eta_{accel,x} \\ y_{accel,y} &= a_y + \eta_{accel,y} \\ y_{accel,z} &= a_z + \eta_{accel,z}, \end{aligned}$$

where a_x , a_y , and a_z represent the acceleration, and $\eta_{accel,x}$, $\eta_{accel,y}$, and $\eta_{accel,z}$ represent the noise, each along its specified axis.

The three rate gyros, also aligned with the body-frame axes, measure rotation around the specific axis,

$$\begin{aligned} y_{gyro,x} &= p + \eta_{gyro,x} \\ y_{gyro,y} &= q + \eta_{gyro,y} \\ y_{gyro,z} &= r + \eta_{gyro,z}, \end{aligned}$$

where $\eta_{gyro,x}$, $\eta_{gyro,y}$, and $\eta_{gyro,z}$ are the noise along each axis.

III.B.2. Radar Odometry

The Radar Odometry approach measures the along-track and cross-track velocity over time. For simplicity, flight is considered in a straight north direction, resulting in

$$\begin{aligned} y_{RO,n} &= \dot{n} + \eta_{RO,n} \\ y_{RO,e} &= \dot{e} + \eta_{RO,e} \\ y_{RO,AGL} &= -\dot{d} + \eta_{RO,AGL}. \end{aligned}$$

III.C. Prediction Models

For the system under consideration, position

$$\mathbf{p} = \begin{bmatrix} n \\ e \\ d \end{bmatrix}$$

and velocity

$$\mathbf{v} = \begin{bmatrix} \dot{n} \\ \dot{e} \\ \dot{d} \end{bmatrix}$$

are represented using the inertial frame, while attitude is represented as

$$\mathbf{\Theta} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix},$$

where ϕ is the roll angle, θ is the pitch angle and ψ is the heading angle.

The system state is given by

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{\Theta} \end{bmatrix}.$$

with the state dynamics

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}).$$

The input for the prediction step is the gyro and accelerometer,

$$\mathbf{u} = \begin{bmatrix} \mathbf{a}^b \\ \omega \end{bmatrix},$$

resulting in the full state transition model

$$f(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \mathbf{v} \\ \mathbf{g} + R_b^i(\mathbf{\Theta}) \mathbf{a}^b \\ S(\mathbf{\Theta}) \omega \end{bmatrix},$$

where $R_b^i(\mathbf{\Theta})$ is the body to inertial frame rotation and $S(\mathbf{\Theta})$ is

$$S(\mathbf{\Theta}) = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix}.$$

The Jacobian of $f(\mathbf{x}, \mathbf{u})$ is given by

$$\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} 0^{3 \times 3} & I^{3 \times 3} & 0^{3 \times 3} \\ 0^{3 \times 3} & 0^{3 \times 3} & \frac{\partial R_b^i(\mathbf{\Theta}) \mathbf{a}^b}{\partial \mathbf{\Theta}} \\ 0^{3 \times 3} & 0^{3 \times 3} & \frac{\partial S(\mathbf{\Theta}) \omega}{\partial \mathbf{\Theta}} \end{bmatrix},$$

where

$$\begin{aligned} \frac{\partial R_b^i(\Theta) \mathbf{a}^b}{\partial \Theta} = & a_x \begin{bmatrix} 0 & -s_\theta c_\psi & -c_\theta s_\psi \\ 0 & -s_\theta s_\psi & c_\theta c_\psi \\ 0 & -c_\theta & 0 \end{bmatrix} + \\ & a_y \begin{bmatrix} c_\phi s_\theta c_\psi + s_\phi s_\psi & s_\phi c_\theta c_\psi & -s_\phi s_\theta s_\psi - c_\phi c_\psi \\ c_\phi s_\theta s_\psi - s_\phi c_\psi & s_\phi c_\theta s_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi \\ c_\phi c_\theta & -s_\phi s_\theta & 0 \end{bmatrix} + \\ & a_z \begin{bmatrix} -s_\phi s_\theta c_\psi + c_\phi s_\psi & c_\phi c_\theta c_\psi & -c_\phi s_\theta s_\psi + s_\phi c_\psi \\ -s_\phi s_\theta s_\psi - c_\phi c_\psi & c_\phi c_\theta s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ -s_\phi c_\theta & -c_\phi s_\theta & 0 \end{bmatrix}, \end{aligned}$$

and

$$\frac{\partial S(\Theta) \omega}{\partial \Theta} = \begin{bmatrix} qc_\phi t_\theta - rs_\phi t_\theta & q\frac{s_\phi}{c_\theta^2} + r\frac{c_\phi}{c_\theta^2} & 0 \\ -qs_\phi - rc_\phi & 0 & 0 \\ q\frac{c_\phi}{c_\theta} - r\frac{s_\phi}{c_\theta} & q\frac{s_\phi s_\theta}{c_\theta^2} + r\frac{c_\phi s_\theta}{c_\theta^2} & 0 \end{bmatrix}.$$

III.D. Radar Odometry Measurement Model

Radar Odometry provides two measurement update models. The AGL update is acquired every 100 milliseconds and may be measured regardless of the measured reflectors. It's update model is of the form

$$h_{RO,AGL}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} -d \end{bmatrix},$$

with the Jacobian

$$\frac{\partial h_{RO,AGL}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

To measure ground velocity, the Radar Odometry update model is

$$h_{RO,V_{ground}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \dot{n} \\ \dot{e} \end{bmatrix},$$

with the associated Jacobian

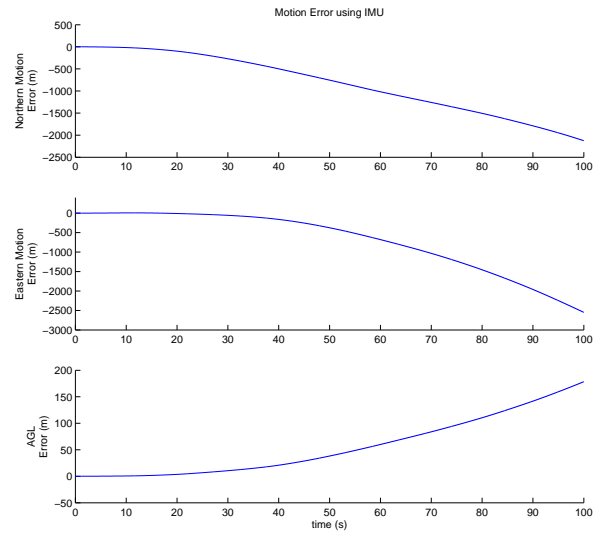
$$\frac{\partial h_{RO,V_{ground}}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The ground velocity measurement is updated as often as the algorithm is able to correctly measure ground motion.

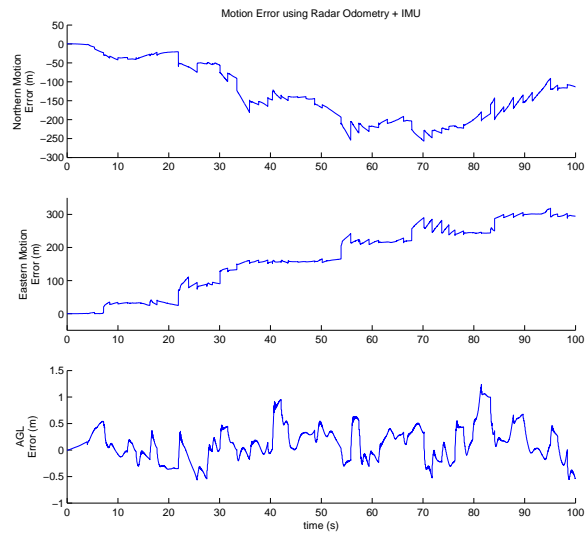
IV. Results

We simulated an unmanned aircraft over an unknown, flat terrain using an IMU alone and using both IMU and Radar Odometry sensors. The simulations used the IMU to propagate the aircraft's pose, while the Radar Odometry algorithm, when used, provided a corrective update term. In all tests, simulated flight dynamics, wind, and sensor noise were implemented. Each test started with an aircraft flying at a fixed, known velocity at a known location.

Figure 11 shows the position error from a 100 second simulation, with the aircraft flying 45 m/s. As expected, the IMU-only solution (Figure 11a) has a large drift rate, resulting in northern position error of up to 2500m, with an eastern position error reaching around 3000m, and AGL error of 200m. Radar Odometry and IMU (Figure 11b) resulted in a significantly smaller drift rate, with worse-case along-track error of 300m, worse-case cross-track error of 300m, and AGL error of less than 2 meters.



a)



b)

Figure 11. Position error over time using a)IMU and b)IMU and RO

V. Conclusion

When the GPS signal is lost or denied, current small UAV systems are unable to accurately estimate their position. Using the Radar Odometry algorithm derived in this paper in simulation has shown to reduce drift to less than 7%, as compared to the 66% drift from an IMU alone. Such results suggest that the using radar for motion estimation is a feasible alternative to traditional vision systems, particularly when using fixed-wing aircraft outdoors. Future work involves handling non-straight flight-tracks and minimizing the error in the current approach. We hope to integrate this solution with other placement recognition work to create a complete solution to GPS-denied navigation.

This work was supported by the Munitions Directorate of the Air Force Research Laboratory under contract FA8651-12-C-0075. Program Manager: Martin Eilders. Principal Investigator: Bryce Ready.

References

- ¹National Magazine Award, Cutting Risk, Rework Through, Year Best, Applying Test, Driven Development, This Week In, Controlling Digital Information, Visit Van Allen Belts, Ryan Smith, and Iraqi Militants Tap. Winner : Radio Eye in the Sky. pages 1–5, 2012.
- ²Richard O Duda and Peter E Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(April 1971):11–15, 1972.
- ³Michael I Duersch. *BYU Micro-SAR: A Very Small, Low-Power LFM-CW Synthetic Aperture Radar*. Master's, Brigham Young University, 2004.
- ⁴F Fraundorfer and D Scaramuzza. Visual Odometry: Part II: Matching, Robustness, Optimization, and Applications. *IEEE Robotics & Automation Magazine*, (June), 2012.
- ⁵Dmitriy Garmatyuk, Jonathan Schuerger, Kyle Kauffman, and Scott Spalding. Wideband OFDM system for radar and communications. *2009 IEEE Radar Conference*, pages 1–6, 2009.
- ⁶Ebi Jose, Martin Adams, Senior Member, John Stephen Mullane, and Nicholas M Patrikalakis. Predicting Millimeter Wave Radar Spectra for Autonomous Navigation. *IEEE Sensors Journal*, 10(5):960–971, 2010.
- ⁷Kyle Kauffman, Yu Morton, John Raquet, and Dmitriy Garmatyuk. Simulation Study of UWB-OFDM SAR for Dead-Reckoning Navigation. In *Institute of Navigation*, pages 153–160, 2010.
- ⁸Kyle Kauffman and John Raquet. Enhanced feature detection and tracking algorithm for UWB-OFDM SAR navigation. *Proceedings of the 2011 IEEE National Aerospace and Electronics Conference (NAECON)*, pages 261–269, July 2011.
- ⁹Dirk Langer. An integrated MMW radar system for outdoor navigation. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, number April, pages 417–422, 1996.
- ¹⁰David Nistér, Oleg Naroditsky, and James Bergen. Visual Odometry. In *Computer Vision and Pattern Recognition 2004*, volume 11, January 2004.
- ¹¹Davide Scaramuzza and Friedrich Fraundorfer. Visual Odometry: Part I The First 30 Years and Fundamentals. *IEEE Robotics & Automation Magazine*, 18(December):80–92, 2011.
- ¹²Bruce Walker and Grant Sander. A high-resolution, four-band SAR Testbed with real-time image formation. In *Geoscience and Remote Sensing Symposium, 1996*, volume 0529, pages 1881–1885, 1996.
- ¹³Richard N. White. Airborne Doppler Radar Navigation of Jet Transport Aircraft. . . and *Navigational Electronics, IRE Transactions on*, pages 11–20, 1962.