Augmenting Inertial Navigation with Image-Based Motion Estimation

Stergios I. Roumeliotis, Andrew E. Johnson and James F. Montgomery[1]

Abstract

Numerous upcoming NASA misions need to land safely and precisely on planetary bodies. Accurate and robust state estimation during the descent phase is necessary. Towards this end, we have developed a new approach for improved state estimation by augmenting traditional inertial navigation techniques with image-based motion estimation (IBME). A Kalman filter that processes rotational velocity and linear acceleration measurements provided from an IMU has been enhanced to accomodate relative pose measurements from the IBME. In addition to increased state estimation accuracy, IBME convergence time is reduced while robustness of the overall approach is improved. The methodology is described in detail and experimental results with a 5DOF gantry testbed are presented.

1 Introduction

NASA's roadmap for solar system exploration is filled with missions that require landing on planets, moons, comets and asteroids. Each mission has its own criteria for success, but all will require some level of autonomous safe and precise landing capability, possibly on hazardous terrain. Previous work [1] has focused on machine vision algorithms that, given a stream of images of a planetary body taken from a single nadir pointing camera, can produce estimates of spacecraft relative motion, spacecraft body absolute position and 3-D topography of the imaged surface. These estimates in turn can be used by spacecraft control systems to follow precise trajectories to planetary surfaces and avoid hazardous terrain while landing.

Our current research, and the focus of this paper, augments inertial navigation techniques [2] with the earlier developed image-based motion estimation (IBME) algorithms [1]. The main contribution of this work is the development of an enhanced Kalman filter (KF) that is able to process *relative pose measurements* from IBME as well as *inertial measurements* from an Inertial Measurement Unit (IMU). This general methodology can be extended to any type of relative measurements (differences between previous and current states of the system). In order to accommodate relative pose measurements, the KF has been modified to duplicate the exact same estimate of the pose of the system and have these two (differently evolving) estimates interact.

The benefits of this approach include increased state estimation accuracy, reduced convergence time of the IBME algorithm and improved robustness of the overall system. Accuracy increases due to additional sensor data for the KF, convergence time decreases because the KF output can be used to initialize the IBME algorithm and robustness improves since the inertial sensors allow continued tracking of spacecraft state even when the IBME fails to follow an adequate number of features. This approach can be used as an enabling technology for future NASA missions that require a safe and precise landing capability.

1.1 Previous Work

The work of Qian et al. [3] is similar to ours in that they fuse both inertial data and imagery using a KF to produce motion estimates. However, we are different in two key aspects. First, we combine both orientation (rate gyros) and position information (accelerometers) with image motion estimates. Qian et al. only fuse orientation information (absolute from magnetometers and rate from gyros) with image motion estimates. Second, our method is meant for an application, safe and precise landing for spacecraft, that is expected to have large shifts in rotation and translation during the spacecraft descent phase. This will drastically effect the field of view after only a few frames, and thus we make no assumption for the reappearence of features in more than two subsequent images. Consequently, our method works by tracking features only between pairs of consecutive images, and not a stream of images as in [3].

Bosse et al. [4] use an Extended Kalman filter to merge IMU, GPS, sonar altimeter, and camera motion estimates into a navigation solution for an autonomous helicopter. Their approach is applicable to a less diverse set of environments for two reasons. First, they use an optical flow based method for motion estimation while we use a correlation based method. Optical flow is restricted to domains where the motion between images is expected to be small, which is not the case for our application. Second, they make the assumption that the surface being imaged is planar, which may not be the case during landing on planetary surfaces. Our method can handle terrain with rugged topography.

Amidi et al. [5] have presented a visual odometer to estimate the position of a helicopter by visually locking on to and tracking ground objects. Attitude information is provided by a set of gyroscopes while position is estimated

[1]S. I. Roumeliotis is with the California Institute of Technology, Pasadena, CA. stergios@robotics.caltech.edu. A. E. Johnson and J. F. Montgomery are with the Jet Propulsion Laboratory, Pasadena, CA. {aej|monty}@robotics.jpl.nasa.gov.

based on template matching from sequences of stereo vision data. In this approach attitude estimates are not derived from the vision algorithm and it is assumed that the field of view changes slowly while the helicopter hovers above the same area. New templates are acquired only when the previous ones dissapear from the scene.

2 Approach

2.1 Inertial sensing and Navigation

Inertial navigation systems (INS) have been used for decades for estimating the position and orientation of manned or autonomous vehicles such as aircrafts [6] and spacecrafts [7, 8]. In recent years, similar systems were employed for localization of autonomous ground vehicles [9, 2]. The core of most INS is an inertial measurement unit (IMU) comprised of 3-axial accelerometers and gyroscopes. Appropriate integration of their signals provides estimates of the location of the vehicle. The quality of these estimates depends primarily on the accuracy and noise profile of the IMU. Such systems can track very accurately sudden motions of short duration but their estimates quickly deteriorate during longer traverses due to the noise that contaminates the IMU signals. More specifically, the integration of the low frequency noise component (bias) in the accelerometer and gyroscope measurements results in tracking errors that grow unbounded with time. In order to sustain positioning accuracy, inertial navigation systems usually include additional sensors (such as compass, inclinometers, star trackers, GPS, deep space network radio signal receivers, etc.) that provide periodic absolute attitude and position measurements. Over the years a variety of approaches have been proposed for optimally combining inertial with absolute measurements. Most current INS employ some form of Kalman filtering for this purpose.

Previously, spacecraft during the entry, descent and landing (EDL) phase of their trip had to rely solely on IMU signals in order to track their position. Lack of absolute position and/or attitude measurements resulted in large positioning errors and therefore left little room for navigation to a desired destination. In addition, tedious postprocessing of their data was necessary for determining their landing site.

In order to improve tracking accuracy during EDL, additional sensing modalities that provide positioning updates are necessary. Cameras are lightweight, space-proven sensors that have been successfully used in the past to provide motion estimates between consecutive image frames [1]. As aforementioned, most INS depend on the availability of *absolute* position and orientation information in order to reduce tracking errors and cannot directly process *relative* pose measurements unless these are expressed as local velocity measurements and processed as such. As it is discussed in detail in [10] this can be problematic, especially in cases where the relative pose measurements are available at a lower rate than the IMU signals. For this reason we have developed a variant of a 6DOF Kalman filter that is capable of optimally fusing inertial measurements from the IMU with displacement estimates provided by a vision-based feature tracking algorithm. At this point we will defer the derivation of this estimator for Section 2.3 and describe first the IBME algorithm.

2.2 Visual Feature Tracking and Navigation

There exist many different types of algorithms for Image-based Motion Estimation (IBME). For efficiency, we use an algorithm that falls in the category of two-frame feature-based motion estimation [11]. To obtain complete 6DOF motion estimates, our algorithm is augmented by altimeter measurements for scale estimation. Below we give a brief overview of our motion estimation algorithm; for more details, please see our previous work [1].

The steps of the algorithm are as follows. At one time instant a descent camera image and a laser altimeter reading are obtained. A short time later, another image and altimeter reading are recorded. The algorithm then processes these pairs of measurements to estimate the rigid 6DOF motion between readings. There are multiple steps in our algorithm. First, distinct features are selected in the first image and then tracked into the second image. Given these feature matches, the motion state and covariance of the spacecraft, up to a scale on translation, are computed. Finally, the magnitude of translation is computed by combining altimetry with the motion, and the motion and motion covariance are scaled accordingly. These measurements are then passed to the Kalman filter where they are combined with inertial measurements to estimate the state of the vehicle.

2.2.1 Feature Detection and Tracking:

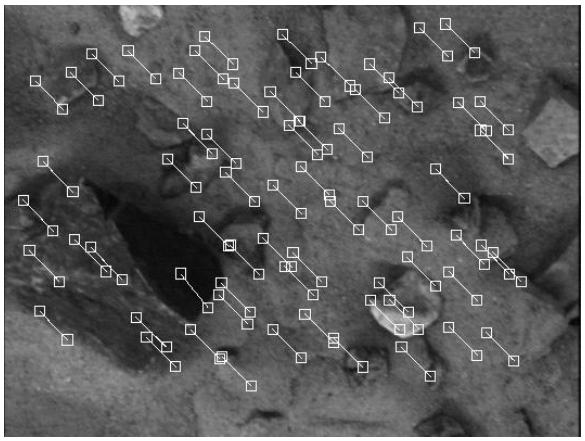

Figure 1: Tracked Features

The first step in two-frame motion estimation is the extraction of features from the first image. Feature detection has been studied extensively and multiple proven feature detection methods exist, so we elected to modify a proven feature detection method instead of redesigning our own. Since processing speed is a very important design constraint for our application, we selected the efficient feature detection algorithm of Benedetti and Perona [12]. This

algorithm is an implementation of the well known Shi-Tomasi feature detector and tracker [13] modified to eliminate transcendental arithmetic. Although they ultimately implemented their algorithm in hardware on a reconfigurable computer, we have found that their algorithmic enhancements also decrease the running time of software implementations.

Usually feature detection algorithms exhaustively search the image for every distinct feature. However, when the goal is motion estimation, only a relatively small number of features need to be tracked (~ 100). Consequently, we can speed up feature detection by using a random search strategy instead of exhaustive search while still guaranteeing that the required number of features is obtained. Suppose that N features are needed for motion estimation. Our detection algorithm selects a pixel at random from the image (uniform distribution in row and column directions). It then computes the sums of intensity derivatives in a neighborhood of the pixel that are used to determine if the pixel is good for tracking. Intensity derivatives are only computed if they have not been computed previously. If the sums of intensity derivatives are large enough (see [12]) then the pixel is selected as a feature for tracking. This procedure is repeated until N features are detected.

Next, features are tracked from the first frame to the second. As with feature detection, there exist multiple methods for feature tracking in the machine vision literature. Feature tracking can be split in to two groups of algorithms: correlation based methods and optical flow based methods. Optical flow feature trackers are appropriate when the motion between images is expected to be small. Since we cannot make this assumption for our application (autonomous EDL for spacecraft), we chose a standard correlation-based feature tracker [14]. This tracker has been made efficient through register arithmetic and sliding window of sums for correlation computation. An example of features selected and tracked between two frames is given in Fig. 1.

2.2.2 Motion Estimation:

The motion between two camera views is described by a rigid transformation $(C(q), t)$ where the rotation $C(q)$, represented as a unit quaternion q, encodes the rotation between views and t encodes the translation between views. Once features are tracked between images, the motion of the camera can be estimated by solving for the motion parameters that, when applied to the features in the first image, bring them close to the corresponding features in the second image.

During estimation the motion parameters are concatenated into a single state vector a

$$a = [q, t] = [q_1, q_2, q_3, q_4, t_x, t_y, t_z]^T \tag{1}$$

Features in the first image are represented in unit focal coordinates as $h_i = [h_{oi}, h_{1i}]$ and in the second image as $h_i' = [h_{oi}', h_{1i}']$. The cost function that is minimized to estimate the motion parameters is

$$L(a) = \sum_i \|h_i^{'} - P(h_i, a)\|^2 \tag{2}$$

where P is the projection of the features in the first image into the second. P is defined as follows. Let X_i be the 3-D coordinate of feature h_i at depth α_i then

$$X_i = [h_{oi}\alpha_i, h_{1i}\alpha_i, \alpha_i]^T \tag{3}$$

The 3-D coordinates of feature $h_i^{'}$ are then

$$X_i^{'} = [x_{i0}^{'}, x_{i1}^{'}, x_{i2}^{'}] = C(q)X_i + t \tag{4}$$

and the projection of h_i into the second image is

$$P(h_i, a) = [x_{i0}^{'}/x_{i2}^{'}, x_{i1}^{'}/x_{i2}^{'}]^T \tag{5}$$

$L(a)$ is minimized using a robust Levenberg-Marquardt nonlinear minimization algorithm, this algorithm requires an initial estimate of the motion parameters and feature depths. In our implementation, the feature depths are all initially set to 1. An initial estimate of the motion parameters can be obtained using any number of methods including: unaided inertial navigation, a robust 8-point algorithm [1], or global search. For the results generated in this paper the initial estimate comes from a simple global search that finds the global minimum of $L(a)$ by searching a coarse discretization of the rotation space, assuming zero translation, and then searching a coarse discretization of the translation space using the best rotation.

During minimization, a robust technique is used to prevent features that are inconsistent with the motion estimate from the previous iteration from corrupting the motion estimate for the current iteration. During each iteration of the minimization, the projection residuals of the features are stored in a vector $R = [r_i, \ldots, r_N]$ and sorted. If $r_i > r_{N/2} + 3(r_{3N/4} - r_{N/4})$ then the corresponding feature is considered an outlier and is removed from the computation of $L(a)$ for the current iteration. After a few iterations (< 10), the estimation converges to the best estimate of the motion $\hat{a}$. Using this estimate, the Fisher information matrix

$$I = \sum_i (dL_i/da * dL_i/da^T) \tag{6}$$

where

$$dL_i/da = -2(dP_i/da)(h_i^{'} - P(h_i, a)) \tag{7}$$

is computed and passed to the Kalman filter for use in generating the covariance of $\hat{a}$.

A fundamental shortcoming of all image-based motion estimation algorithms is the inability to solve for the magnitude of translational motion $\|t\|$, so the output of motion estimation is a 5DOF motion composed of a unit vector describing the direction of heading and the rotation between views. As described in [1] laser altimetry can be combined with the 5DOF motion estimate to compute the complete 6DOF motion of the vehicle. However, for the gantry results presented in this paper, the altimetry data was too

coarse, so the magnitude of translation was derived from the change in gantry positions between image captures.
For each image pair, the output sent to the Kalman filter for image-based motion estimation is the relative pose measurement $[z_p \;\; z_q]^T$ and its corresponding covariance R_r calculated based on the motion state a, the magnitude of translation $\|t\|$ and the Fisher information matrix I.

2.3 Fusion of Inertial and Relative Sensor data

In this section we derive the equations for the modified Kalman filter that processes the relative pose measurements from the IBME algorithm. Since our formulation is based on sensor modeling, we use the Indirect form of the Kalman filter that estimates the errors in the estimated states instead of the states themselves. The interested reader is referred to [15, 2, 16] for a detailed description of the advantages of the Indirect KF vs. the Direct KF.
In what follows, we assume that at time t_k the vehicle is at position ${}^G p(t_k) = {}^G p_1$ with (quaternion) attitude ${}^1_G q(t_k) = q_1$ and after m steps it has moved to position ${}^G p(t_{k+m}) = {}^G p_2$ with attitude ${}^2_G q(t_{k+m}) = q_2$. Frames $\{G\}$, $\{1\}$, and $\{2\}$ are the inertial frames of reference attached to the vehicle at times t_0, t_k and t_{k+m} correspondingly.

2.3.1 Relative Position Measurement Error:

The relative position measurement z_p between the two locations $\{1\}$, and $\{2\}$ can be written as:

$$z_p = {}^1 p_2 + n_p = {}^G_1 C^T(q)({}^G p_2 - {}^G p_1) + n_p \tag{8}$$

where n_p is the noise associated with this measurement assumed to be a zero-mean white Gaussian process with covariance $R_p = E[n_p n_p^T]$. ${}^G_1 C(q)$ is the rotational matrix that expresses the orientation transformation between frames $\{G\}$ and $\{1\}$.
If Δp_i is the error in the estimate of position p_i and δq is the error in the estimate of attitude q then:[1]

$$p_i = \Delta p_i + \widehat{p}_i,\ i = 1, 2.\ ,\ \ q = \delta q \otimes \widehat{q}$$

Equation (8) can now be written as:

$$z_p = {}^G_1 C^T(\delta q \otimes \widehat{q})({}^G \widehat{p}_2 + \Delta p_2 - {}^G \widehat{p}_1 - \Delta p_1) + n_p \tag{9}$$

The estimated relative position measurement is:

$$\widehat{z}_p = {}^G_1 C^T(\widehat{q})({}^G \widehat{p}_2 - {}^G \widehat{p}_1) = {}^G_1 C^T(\widehat{q})\, {}^G \widehat{p}_{1,2} \tag{10}$$

The error in the relative position measurement is:

$$\Delta z_p = z_p - \widehat{z}_p$$

By substituting from Eqs. (9), (10) and employing the small error angle approximation $\delta q \simeq [\delta \vec{q} \;\; 1]^T \simeq \left[\frac{1}{2}\vec{\theta}_1 \;\; 1\right]^T$, it can be shown [10] that:

$$\begin{aligned} \Delta z_p \;\; \simeq \;\; & {}^G_1 C^T(\widehat{q}_1)\lfloor {}^G \widehat{p}_{1,2} \rfloor \delta\vec{\theta}_1 + \\ & {}^G_1 C^T(\widehat{q}_1)\Delta p_2 - {}^G_1 C^T(\widehat{q}_1)\Delta p_1 + n_p \end{aligned} \tag{11}$$

[1] Note that from here on q refers to q_1 and δq refers to δq_1. We have also dropped the vector symbol from the real, measured, estimated and error position to simplify notation.

where $\lfloor \;\; \rfloor$ denotes the cross product matrix of a vector

$$\lfloor \vec{V} \rfloor = \begin{bmatrix} 0 & -V_3 & V_2 \\ V_3 & 0 & -V_1 \\ -V_2 & V_1 & 0 \end{bmatrix}$$

In Eq. (11) the first term expresses the effect of the orientation uncertainty at time t_k on the quality of the estimated measurement. Note that if at time t_k there was no uncertainty about the orientation of the vehicle that would mean $\delta\vec{\theta}_1 = 0$ and thus the error in the relative position measurement would depend only on the errors in the estimates of the previous and current position of the vehicle.

2.3.2 Relative Attitude Measurement Error:

The relative attitude measurement error between the two locations $\{1\}$, and $\{2\}$ is:

$$\Delta z_q = z_q - {}^1_2\widehat{q} = {}^1_2 q + n_q - {}^1_2\widehat{q} \tag{12}$$

where n_q is the relative attitude measurement noise. We assume that n_q is a zero-mean white Gaussian process with covariance $R_q = E[n_q n_q^T]$. Since

$${}^1_2 q = {}^1_0 q \otimes {}^2_0 q^{-1} = q(t_k) \otimes q^{-1}(t_{k+m}) = q_1 \otimes q_2^{-1}$$

and

$$q_i = \delta q_i \otimes \widehat{q}_i\ ,\ i = 1, 2.$$

${}^1_2 q$ can be written as:

$${}^1_2 q = \delta q_1 \otimes {}^1_2\widehat{q} \otimes \delta q_2^{-1} \tag{13}$$

By substituting Eq. (13) in Eq. (12) we have:

$$\Delta z_q = \delta q_1 \otimes {}^1_2\widehat{q} \otimes \delta q_2^{-1} - {}^1_2\widehat{q} + n_q \tag{14}$$

with

$$\delta q_1 = \begin{bmatrix} \delta\vec{q}_1 \\ \delta q_{1_4} \end{bmatrix},\ \delta q_2 = \begin{bmatrix} \delta\vec{q}_2 \\ \delta q_{2_4} \end{bmatrix},\ \delta q_2^{-1} = \begin{bmatrix} -\delta\vec{q}_2 \\ \delta q_{2_4} \end{bmatrix}$$

In order to simplify the notation we set:

$${}^1_2\widehat{q} = q = \begin{bmatrix} \vec{q} \\ q_4 \end{bmatrix}$$

For small attitude estimation errors δq_1 and δq_2 we make the following approximations: $\delta q_{1_4} \simeq 1, \delta q_{2_4} \simeq 1, \delta\vec{q}_1 << \mathbf{1}_{3\times 1}, \delta\vec{q}_2 << \mathbf{1}_{3\times 1}$. The first term in Eq. (14) can be written as:

$$\delta q_1 \otimes {}^1_2\widehat{q} \otimes \delta q_2^{-1} \simeq \begin{bmatrix} \vec{q} + q_4(\delta\vec{q}_1 - \delta\vec{q}_2) + \lfloor\vec{q}\rfloor(\delta\vec{q}_1 + \delta\vec{q}_2) \\ q_4 - \vec{q}^T(\delta\vec{q}_1 - \delta\vec{q}_2) \end{bmatrix} \tag{15}$$

By multiplying both sides of Eq. (12) with the matrix

$$\Xi^T({}^1_2\widehat{q}) = \Xi^T(q) = \begin{bmatrix} (q_4 I - \lfloor\vec{q}\rfloor) & -\vec{q} \end{bmatrix} \tag{16}$$

we define the *vector* attitude measurement error as:

$$\begin{aligned} \Delta\tilde{z}_q \;\; &= \;\; \Xi^T({}^1_2\widehat{q})\Delta z_q = \Xi^T({}^1_2\widehat{q})({}^1_2 q + n_q) - 0 \\ &= \;\; \Xi^T({}^1_2\widehat{q})(\delta q_1 \otimes {}^1_2\widehat{q} \otimes \delta q_2^{-1}) + \Xi^T({}^1_2\widehat{q}) n_q \end{aligned} \tag{17}$$

By substituting from Eqs. (15), (16) the first term in the previous equation can be written as:

$$\Xi^T(\tfrac{1}{2}\widehat{q})(\delta q_1 \otimes \tfrac{1}{2}\widehat{q} \otimes \delta q_2^{-1}) \simeq \delta\vec{q}_1 - \tfrac{1}{2}C(\tfrac{1}{2}\widehat{q})\delta\vec{q}_2 \qquad (18)$$

Eq. (17) is now expressed as:

$$\Delta\tilde{z}_q \simeq \frac{1}{2}(\delta\vec{\theta}_1 - \tfrac{1}{2}C(\tfrac{1}{2}\widehat{q})\delta\vec{\theta}_2) + \tilde{n}_q \qquad (19)$$

where we have used the small angle approximation $\delta\vec{q}_i = \frac{1}{2}\vec{\theta}_i$, $i = 1, 2$ and $\tilde{n}_q = \Xi^T(\frac{1}{2}\widehat{q})n_q$ with

$$\tilde{R}_q = E[\tilde{n}_q\tilde{n}_q^T] = \Xi^T(\tfrac{1}{2}\widehat{q})R_q\Xi(\tfrac{1}{2}\widehat{q})$$

2.3.3 Relative Pose Measurement Error:

The Indirect Kalman filter estimates the errors in: (i) attitude $\delta\vec{\theta}$, (ii) gyroscopes biases $\Delta\vec{b}_g$, (iii) velocity $\Delta\vec{v}$, (iv) accelerometers biases $\Delta\vec{b}_a$, and (v) position $\Delta\vec{p}$. The error state vectors estimated by the filter at times t_k and t_{k+m} for $i = 1, 2$ are:

$$\Delta x_i = \left[\begin{array}{ccccc} \delta\vec{\theta}_i^{\,T} & \Delta\vec{b}_{gi}^{\,T} & \Delta\vec{v}_i^{\,T} & \Delta\vec{b}_{ai}^{\,T} & \Delta\vec{p}_i^{\,T} \end{array}\right]^T$$

The errors in the relative position and attitude (pose) measurements calculated in Eqs. (11) and (19) are:

$$\begin{aligned} \Delta\tilde{z}_{k+m} &= \left[\begin{array}{c} \Delta z_p \\ \Delta\widetilde{z}_q \end{array}\right] = \mathcal{X}\left[\begin{array}{c} \Delta z_p \\ \Delta z_q \end{array}\right] \qquad (20) \\ &= \Gamma\,[D_1\ D_2]\left[\begin{array}{c} \Delta x_1 \\ \Delta x_2 \end{array}\right] + \mathcal{X}\ n_r = H\left[\begin{array}{c} \Delta x_1 \\ \Delta x_2 \end{array}\right] + \widetilde{n}_r \end{aligned}$$

with

$$\begin{aligned} \Gamma &= \left[\begin{array}{cc} {}^G_1C^T(\widehat{q}_1) & 0 \\ 0 & {}^G_1C^T(\widehat{q}_1) \end{array}\right] \\ D_1 &= \left[\begin{array}{ccccc} \lfloor {}^G\widehat{p}_{1,2}\rfloor & 0 & 0 & 0 & -I \\ (\frac{1}{2})\,{}^G_1C(\widehat{q}_1^{-1}) & 0 & 0 & 0 & 0 \end{array}\right] \\ D_2 &= \left[\begin{array}{ccccc} 0 & 0 & 0 & 0 & I \\ (-\frac{1}{2})\,{}^G_2C(\widehat{q}_2^{-1}) & 0 & 0 & 0 & 0 \end{array}\right] \\ \mathcal{X} &= \left[\begin{array}{cc} I_{3\times3} & 0_{3\times4} \\ 0_{3\times3} & \Xi^T(\frac{1}{2}\widehat{q}) \end{array}\right] \end{aligned}$$

Both noise vectors n_r and $\widetilde{n}_r$ are assumed to be zero-mean white Gaussian processes with

$$R_r = E\left[n_r n_r^T\right] = \left[\begin{array}{cc} R_p & R_{pq} \\ R_{qp} & R_q \end{array}\right],\ \widetilde{R}_r = E[\widetilde{n}_r\widetilde{n}_r^T] = \mathcal{X}R_r\mathcal{X}^T$$

As it is evident from Eq. (21), the relative pose measurement error is expressed in terms of the current $\Delta x_2 = \Delta x(t_{k+m})$ and the previous $\Delta x_1 = \Delta x(t_k)$ (error) state of the system. Therefore the Kalman filter state vector has to be appropriately augmented to contain both these state estimates. Note that t_k and t_{k+m} are the time instants when the two images processed by the IBME were recorded and thus the relative pose (motion estimate) measurement provided by the IBME corresponds to the time interval $[t_k, t_{k+m}]$.

2.3.4 Augmented-state propagation:

If $\Delta x_{k/k}$ is the state estimate at time t_k (when the first image was recorded) we augment the state vector with a second copy of this estimate:

$$\Delta\breve{x} = \left[\begin{array}{cc} \Delta x_{k/k}^T & \Delta x_{k/k}^T \end{array}\right]^T$$

Since initially, at time t_k, both versions of the estimate of the error state contain the same amount of information, the covariance matrix for the augmented system would be:

$$\breve{P}_{k/k} = \left[\begin{array}{cc} P_{kk} & P_{kk} \\ P_{kk} & P_{kk} \end{array}\right]$$

where P_{kk} is the covariance matrix for the (error) state of the vehicle at time t_k. In order to conserve the estimate of the state at t_k, necessary for evaluating the relative pose measurement error at t_{k+m}, during this interval only the second copy of the state estimate is propagated while the first one remains stationary.[2] The propagation equation for the augmented system based on Eq. (26) is:

$$\left[\begin{array}{c} \Delta x_1 \\ \Delta x \end{array}\right]_{k+1/k} = \left[\begin{array}{cc} I & 0 \\ 0 & F_{k+1} \end{array}\right]\left[\begin{array}{c} \Delta x_1 \\ \Delta x \end{array}\right]_{k/k} + \left[\begin{array}{c} 0 \\ G_{k+1} \end{array}\right]\vec{w}_k$$

or

$$\Delta\breve{x}_{k+1/k} = \breve{F}_{k+1}\Delta\breve{x}_{k/k} + \breve{G}_{k+1}\vec{w}_k$$

where Δx_1 is the non-moving copy of the error state of the vehicle.[3] The covariance of the augmented system is propagated according to Eq. (27) and after m steps is:

$$\breve{P}_{k+m/k} = \left[\begin{array}{cc} P_{kk} & P_{kk}\mathcal{F}^T \\ \mathcal{F}P_{kk} & P_{k+m/k} \end{array}\right] \qquad (21)$$

where $\mathcal{F} = \prod_{i=1}^{m} F_{k+i}$ and $P_{k+m/k}$ is the propagated covariance of the evolving state at time t_{k+m}.

2.3.5 Update equations:

When the relative pose measurement is received the covariance matrix for the residual is given by Eq. (28):

$$S = H\breve{P}_{k+m/k}H^T + \widetilde{R}_r \qquad (22)$$

where $\widetilde{R}_r = \mathcal{X}R_r\mathcal{X}^T$ is the adjusted covariance for the the relative pose measurement and R_r is the initial covariance of this noise as calculated by the IBME algorithm. We define the pseudo-residual covariance matrix as $\widetilde{S} = \Gamma^{-1}S\Gamma$ and by substituting from Eqs. (21), (22):

$$\widetilde{S} = D_1P_{kk}D_1^T + D_2\mathcal{F}P_{kk}D_1^T + D_1P_{kk}\mathcal{F}^TD_2^T + D_2P_{k+m/k}D_2^T + \mathcal{R}_r$$

[2] In the derivation of the equations of the Kalman filter that processes relative pose measurements, we duplicated the state estimate and its corresponding covariance at time t_k and allowed each of them to evolve separately. We have coined the term *stochastic cloning* for this new technique. From the Oxford English Dictionary: [ad. Greek $\kappa\lambda\acute{\omega}\nu$ twig, slip], [*clone, v.* To propagate or reproduce (an identical individual) from a given original; to replicate (an existing individual).]

[3] The discrete time state and system noise propagation matrices F_{k+1}, G_{k+1} are described in detail in [17, 16, 10].

where $\mathcal{R}_r = \Gamma^{-1}\widetilde{R}_r\Gamma$. The updated covariance matrix is calculated from Eq. (30) as:

$$\begin{aligned}\breve{P}_{k+m/k+m} &= \breve{P}_{k+m/k} - \breve{P}_{k+m/k}H^T S^{-1} H \breve{P}_{k+m/k} \qquad (23)\\ &= \breve{P}_{k+m/k} - \begin{bmatrix} P_{kk}D_1^T + P_{kk}\mathcal{F}^T D_2^T \\ \mathcal{F}P_{kk}D_1^T + P_{k+m/k}D_2^T \end{bmatrix} \widetilde{S}^{-1} \times \\ & \quad \begin{bmatrix} D_1 P_{kk} + D_2\mathcal{F}P_{kk} & D_1 P_{kk}\mathcal{F}^T + D_2 P_{k+m/k} \end{bmatrix}\end{aligned}$$

The updated covariance matrix for the new state of the vehicle will be (lower-right diagonal submatrix):

$$\begin{aligned}&P_{k+m/k+m} = P_{k+m/k} - \\ &(\mathcal{F}P_{kk}D_1^T + P_{k+m/k}D_2^T)\tilde{S}^{-1}(D_1 P_{kk}\mathcal{F}^T + D_2 P_{k+m/k})\end{aligned}$$

The Kalman gain is calculated by applying Eq. (29):

$$K = \begin{bmatrix} K_1 \\ K_2 \end{bmatrix} = \breve{P}_{k+m/k}H^T S^{-1} \qquad (24)$$

with

$$K_2 = (\mathcal{F}P_{kk}D_1^T + P_{k+m/k}D_2^T)\tilde{S}^{-1}\Gamma^T \qquad (25)$$

The residual is calculated as in Eq. (31):

$$\tilde{r}_{k+m} = \Delta\tilde{z}_{k+m} = \mathcal{X}\Delta z_{k+m} = \begin{bmatrix} z_p - \hat{z}_p \\ \Xi^T(\widehat{{}^1_2 q})(z_q - \hat{z}_q) \end{bmatrix}$$

where z_p, z_q are the relative position and orientation measurements provided by the IBME,

$$\hat{z}_p = {}^1\hat{p}_2 = {}^G_1C^T(\hat{q}_1)({}^G\hat{p}_2 - {}^G\hat{p}_1)\ , \quad \hat{z}_q = {}^1_2\hat{q}$$

and

$$\Xi^T(\widehat{{}^1_2 q})\hat{z}_q = \Xi^T(\widehat{{}^1_2 q})\,{}^1_2\hat{q} = 0_{3\times 1}$$

Thus

$$\tilde{r}_{k+m} = \begin{bmatrix} z_p - {}^G_1C^T(\hat{q}_1)\,{}^G\hat{p}_{1,2} \\ \Xi^T(\widehat{{}^1_2 q})z_q \end{bmatrix}$$

Finally, the updated augmented state is given by Eq. (32):

$$\breve{x}_{k+m/k+m} = \breve{x}_{k+m/k} + K\tilde{r}_{k+m}$$

From Eq. (25) the (evolving) state will be updated as:

$$\begin{aligned}x_{k+m/k+m} &= x_{k+m/k} + (\mathcal{F}P_{kk}D_1^T + P_{k+m/k}D_2^T)\tilde{S}^{-1} \times \\ &\quad \left(\mathcal{Z}_{k+m} - \begin{bmatrix} {}^G\hat{p}_{1,2} \\ 0 \end{bmatrix}\right)\end{aligned}$$

where

$$\mathcal{Z}_{k+m} = \begin{bmatrix} {}^G_1C(\hat{q}_1^{-1})z_p \\ {}^G_1C(\hat{q}_1^{-1})\Xi^T(\widehat{{}^1_2 q})z_q \end{bmatrix}$$

is the pseudo-measurement of the relative displacement (pose) expressed in global coordinates. The quantities $\hat{q}_1^{-1}$, ${}^1_2\hat{q} = {}^1_G\hat{q} \otimes {}^2_G\hat{q}^{-1} = \hat{q}_1 \otimes \hat{q}_2^{-1}$, and $\hat{p}_{1,2} = {}^G\hat{p}_2 - {}^G\hat{p}_1$ are computed using the previous and current state estimates from the filter. Note that the current state estimates at time t_{k+m} are calculated by propagating the previous state estimates at time t_k using the rotational velocity and linear acceleration measurements from the IMU.

The same process is repeated every time a new set of relative pose measurements $z(t_{k+\lambda m}) = \left[z_p^T(t_{k+\lambda m})\ \ z_q^T(t_{k+\lambda m})\right]^T, \lambda = 1, 2, ..$ becomes available. The previous treatment makes the assumption that the measurements $z(t_{k+\lambda m})$ are mutually independent, i.e. $E\{z(t_{k+\lambda_i m})z^T(t_{k+\lambda_j m})\} = 0$. If the IBME algorithm uses the same set of features from an intermediate image to track the pose of the vehicle through two consecutive steps then these measurements are loosely correlated:

$$\begin{aligned}E\{z(t_{k+\lambda m})z^T(t_{k+(\lambda+1)m})\} &\neq 0\\ E\{z(t_{k+\lambda m})z^T(t_{k+(\lambda+1)m})\} &<< E\{z(t_{k+\lambda m})z^T(t_{k+\lambda m})\}\end{aligned}$$

In this case the correlations have to be explicitly addressed by the estimation algorithm. The interested reader is referred to [18] for a detailed treatment of this case.

3 Experimental Results

3.1 Gantry description

Experiments were performed on a 5DOF gantry testbed at JPL, shown in Fig. 2. The gantry provides a hardware in the loop testbed for collecting data of a simulated planetary surface useful for validating algorithms in a controlled environment. It can be commanded to move linearly at constant velocity in x, y, and z and can also pan and tilt. The gantry provides ground truth linear measurements with 0.35 millimeter resolution and angular measurements with 0.01 degree resolution at up to a 4 Hz data rate. It can carry payloads weighing up to 4 pounds and still pan and tilt while heavier payloads can be carried with linear motions only. For our experiments, we attached an

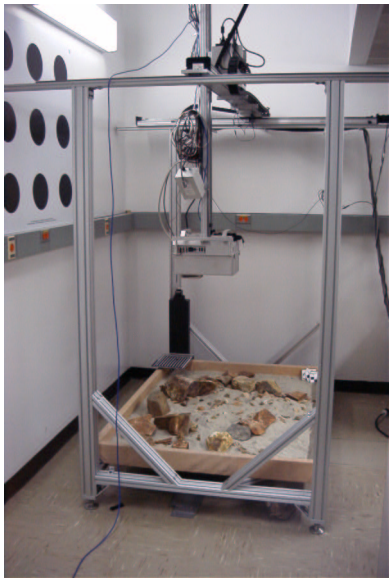

Figure 2: 5DOF Gantry Testbed

electronics package containing sensors, onboard computers, and battery power to the gantry. The sensors include a Crossbow DMU-VGX IMU, a downward pointing laser altimeter and greyscale CCD camera. Only the IMU and CCD camera were used for these experiments. The laser

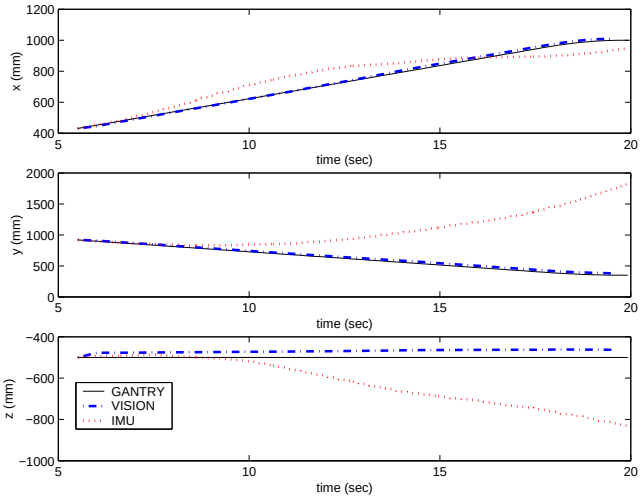

Figure 3: Trajectories estimated by integration of (i) the relative pose measurements provided by the IBME vision-based algorithm, (ii) the IMU signals, and (iii) the gantry.

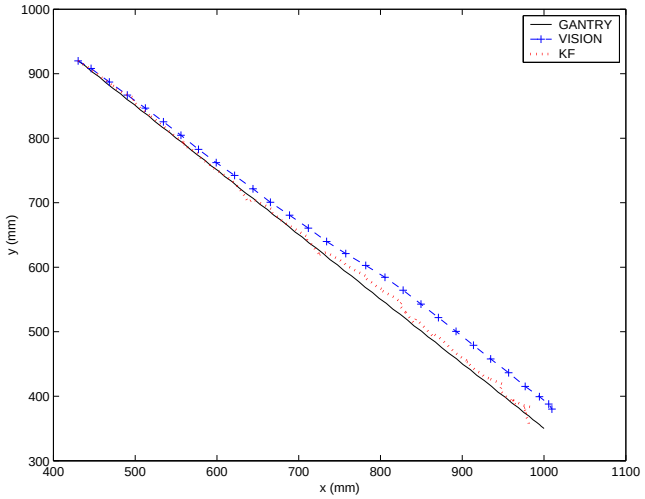

Figure 4: Trajectories estimated by (i) the integration of the relative pose measurements provided by the IBME vision-based algorithm, (ii) the Kalman filter, and (iii) the gantry (x-y).

altimeter will be integrated into the approach in the future. For our experiments, only linear motions were commanded, pan and tilt was held at zero degrees orientation due to weight constraints as described above. All sensor and ground truth gantry data were gathered while following these trajectories and then processed and analyzed off-line.

3.2 Preliminary results

In the results presented here the motion of the vehicle is tracked after it has been accelerated to a speed of $v = [42.7 \quad -42.7 \quad 0]^T \, mm/sec$ at $t = 5.5sec$. For the rest of the time the vehicle follows a straight line, almost constant velocity, trajectory until time $t = 18sec$ when it decelerates to a full stop at $t = 19.6sec$. In order to extract the actual body acceleration during this motion, the local projection of the gravitational acceleration vector ${}^L\vec{g} = {}^G_L C(q)^T {}^G\vec{g}$ has to be subtracted from the accelerometer signals. Even small errors in the attitude estimate $\hat{q}$ can cause significant errors in the calculation of the actual body accelerations. This is more prevalent during slow motions with small body accelerations as the ones during this experiment. The estimated velocities and positions through the integration of the IMU are susceptible to large errors due to the magnitude of the gravitational acceleration compared to the minute body accelerations that the vehicle experiences during its actual motion. With ${}^L\vec{g}$ being the dominant acceleration in the measured signals, error analysis based on the propagation equations of the IMU integrator has shown that even for small attitude errors such as $\delta\theta = 1^\circ$ the errors in position can grow as $\Delta p_x \simeq \|g\| \; \delta\theta \; t^2 = 171 \; t^2 \; mm$ while the vehicle only moves as $p_x \simeq v_x \; t = 42.7 \; t \; mm$. This is evident in Fig. 3 where the position estimates calculated by appropriate integration of the IMU signals are valid for only a short period of time before the errors grow unbounded (e.g. for the y component of position the final error is over $1500mm$). Note also that during this relatively slow motion the integration of the IBME estimates provides significantly better results with only small errors introduced at the begining of the experiment (end of the acceleration phase).

By combining the IBME relative pose measurements with the IMU signals within the Kalman filter, positioning errors are substantially reduced. The vehicle trajectory estimated by the KF follows closer the ground truth path (recorded by the gantry) when compared to the trajectories estimated previously by either the IMU or the IBME. The position estimates for $(x - y)$ and $(z - t)$ are shown in Figs. (4) and (5a) respectively. The average (absolute) errors in these estimates were $|\,\overline{\Delta p}\,| = [4.5 \quad 4.7 \quad 4.2]^T \; mm$ for the KF, $|\,\overline{\Delta p}\,| = [17.4 \quad 41.4 \quad 29.9]^T \; mm$ for the IBME and $|\,\overline{\Delta p}\,| = [53.5 \quad 464.7 \quad 126.1]^T \; mm$ for the IMU.

The availability of intermittent (relative) positioning information enables the filter to also update the estimates of the biases in the accelerometer and gyroscope signals as depicted in Fig. (5b). This in effect reduces the errors in the linear acceleration and rotational velocity measurements and allows the KF estimator to operate for longer periods of time before an external absolute pose measurement is necessary. Finally, we should note that since the information from the IBME corresponds to *relative* and not *absolute* pose measurements the filter estimated errors will continue to grow, albeit at a slower rate. This rate is determined by the frequency and quality of the IBME measurements.

4 Summary and Future Work

The motivation for this work is to advance the state of the art in technologies that enable autonomous safe and precise landing on planetary bodies. However, the general Kalman filter methodology described here can be

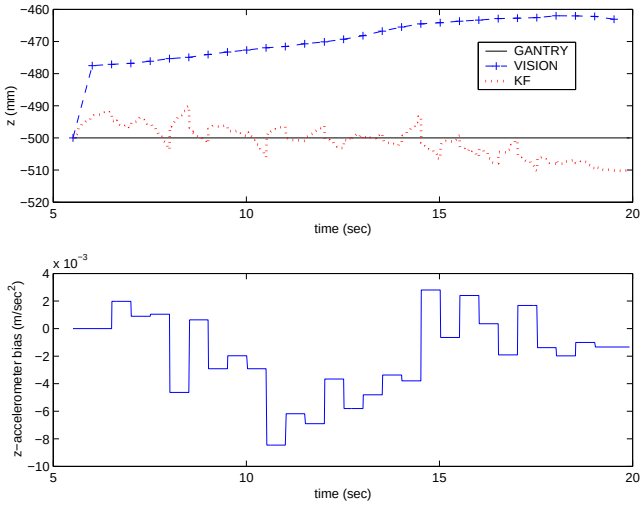

Figure 5: (a) Trajectories estimated by (i) the integration of the relative pose measurements provided by the IBME vision-based algorithm, (ii) the Kalman filter, and (iii) the gantry (z-t). (b) z-axis accelerometer bias estimates.

extended to any application that combines traditional inertial sensors with those that provide relative measurements (differences between previous and current states of the system).

We have shown that augmenting inertial navigation with image-based motion estimation provides advantages over using each method separately by increasing state estimation accuracy. In addition, the fusion of IMU measurements with motion estimates from the IBME increases the robustness of the Kalman filter estimator due to the fact that these two sensing modalities have complimentary operating conditions and noise characteristics. The IMU is capable of tracking sudden motions but drifts fast during longer smoother trajectories while the IBME is best suited for low frequency movements and is immune to low frequency drifts. Finally, although propagated pose estimates from the KF were not used to initialize IBME at this stage, we expect that this will increase the speed of convergence of the IBME algorithm by reducing the size of the search space.

Acknowledgments

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. We would like to thank Chuck Bergh for his work on the gantry and the electronics package.

A Indirect Kalman Filter equations

Propagation

$$\Delta x_{k+1/k} = F_{k+1}\Delta x_{k/k} + G_{k+1}w_k \tag{26}$$

$$P_{k+1/k} = F_{k+1}P_{k/k}F_{k+1}^T + G_{k+1}Q_kG_{k+1}^T \tag{27}$$

Update

$$S = HP_{k+1/k}H^T + R \tag{28}$$

$$K = P_{k+1/k}H^TS^{-1} \tag{29}$$

$$P_{k+1/k+1} = P_{k+1/k} - P_{k+1/k}H^TS^{-1}HP_{k+1/k} \tag{30}$$

$$r_{k+1} = z_{k+1} - \widehat{z}_{k+1} = \Delta z_{k+1} \tag{31}$$

$$x_{k+1/k+1} = x_{k+1/k} + Kr_{k+1} \tag{32}$$

References

[1] A. E. Johnson and L. H. Matthies, "Precise image-based motion estimation for autonomous small body exploration," in *Proc. 5th Int'l Symp. On Artificial Intelligence, Robotics and Automation in Space*, Noordwijk, The Netherlands, June 1-3 1999, pp. 627–634.

[2] S. I. Roumeliotis, G. S. Sukhatme, and G. A. Bekey, "Circumventing dynamic modeling: Evaluation of the error-state Kalman filter applied to mobile robot localization," in *Proceedings of IEEE International Conference on Robotics and Automation*, Detroit, MI, May 10-15 1999, vol. 2, pp. 1656–1663.

[3] G. Qian, R. Chellappa, Q. Zheng, and J. Ortolf, "Camera motion estimation using monocular image sequences and inertial data," Tech. Rep. CS-TR-3997, University of Maryland, College Park, MD, 1999.

[4] M. Bosse, W.C. Karl, D. Castanon, and P. DiBitetto, "A vision augmented navigation system," in *Proceedings of IEEE Conference on Intelligent Transportation Systems*, Boston, MA, 9-12 Nov. 1997, pp. 1028–33.

[5] O. Amidi, T. Kanade, and K. Fujita, "A visual odometer for autonomous helicopter flight," *Robotics and Autonomous Systems*, vol. 28, no. 2-3, pp. 185–193, Aug. 1999.

[6] J. A. Farrell, T. D. Givargis, and M. J. Barth, "Real-time differential carrier phase GPS-aided INS," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 4, pp. 709–721, July 2000.

[7] J. R. Wertz, Ed., *Spacecraft Attitude Determination and Control*, vol. 73 of *Astrophysics and Space Science Library*, D. Reidel Publishing Company, Dordrecht, The Netherlands, 1978.

[8] E. J. Lefferts, F. L. Markley, and M. D. Shuster, "Kalman filtering for spacecraft attitude estimation," *Journal of Guidance, Control, and Dynamics*, vol. 5, no. 5, pp. 417–429, Sept.-Oct. 1982.

[9] S. Sukkarieh, E. M. Nebot, and H. F. Durrant-Whyte, "A high integrity IMU/GPS navigation loop for autonomous land vehicle applications," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 572–578, June 1999.

[10] S. I. Roumeliotis, "A Kalman filter for processing 3-D relative pose measurements," Tech. Rep., Robotics Laboratory, California Institute of Technology, Sep. 2001, http://robotics.caltech.edu/~stergios/tech_reports/relative_3d_kf.pdf.

[11] J. Weng, N. Ahuja, and T. Huang, "Optimal motion and structure estimation," *IEEE Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 864–884, 1993.

[12] A. Benedetti and P. Perona, "Real-time 2-D feature detection on a reconfigurable computer," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Santa Barbara, CA, 23-25 June 1998, pp. 586–593.

[13] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'94)*, Seattle, WA, 21-23 June 1994, pp. 593–600.

[14] L. Matthies, *Dynamic Stereo Vision*, Ph.D. thesis, School of Computer Science, Carnegie Mellon University, 1989.

[15] E. J. Lefferts and F. L. Markley, "Dynamics modeling for attitude determination," *AIAA Paper 76-1910*, Aug. 1976.

[16] Stergios I. Roumeliotis, *Robust Mobile Robot Localization: From single-robot uncertainties to multi-robot interdependencies*, Ph.D. thesis, Electrical Engineering Department, University of Southern California, Los Angeles, CA, May 2000.

[17] B. Friedland, "Analysis strapdown navigation using quaternions," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-14, no. 5, pp. 764–768, Sep. 1978.

[18] S. I. Roumeliotis, "Treatment of correlations in relative pose measurements," Tech. Rep., Robotics Laboratory, California Institute of Technology, March 2002, http://robotics.caltech.edu/~stergios/tech_reports/tr_rpm_correl.pdf.