Orbital SLAM

Corinne Vassallo
Dept of Aerospace Engineering & Engineering Mechanics
The University of Texas at Austin
Austin, TX, USA
corinne.vassallo@utexas.edu

Wennie Tabib
Astrobotic Technology, Inc
Pittsburgh, PA, USA
wennie.tabib@astrobotic.com

Kevin Peterson
Astrobotic Technology, Inc
Pittsburgh, PA, USA
kevin.peterson@astrobotic.com

***Abstract*—This paper demonstrates infrastructure-free orbital Simultaneous Localization and Mapping (SLAM). Individual surface landmarks are tracked through images taken in orbit and the filter receives measurements of these landmarks in the form of bearing angles. The filter then updates the spacecraft's position and velocity as well as landmark locations, thus building a map of the orbited body. In contrast to other approaches that use an IMU, which doesn't work in orbit, to resolve scale, the contribution of this paper is to demonstrate that scale can be resolved using orbital dynamics. Radio localization can be replaced with onboard localization, enabling truly autonomous missions to both under-mapped and unmapped planetary bodies.**

Overall system convergence is shown by simulating landmark detection from an orbit of the Clementine Mission on a Moon model constructed using Lunar Reconnaissance Orbiter (LRO) digital elevation data in conjunction with the filter. The techniques developed in this work demonstrate that when combined with a gravity model, visual SLAM converges to a full scale solution.

***Keywords*-orbit determination; slam; landmark detection; bearing angles**

I. Introduction

This paper presents an approach to visual simultaneous localization and mapping (SLAM) for spacecraft in orbit around a planetary body or asteroid. The approach specifically takes advantage of the knowledge of the gravity vector to determine scale. Landmarks on the orbited body are tracked in a sequence of camera images. Measurements are received by the spacecraft filter in the form of bearing angles. These measurements combined with knowledge of orbital mechanics are used to update spacecraft position and velocity as well as landmark position.

Traditional SLAM approaches for ground vehicles and UAVs have used inertial data to resolve for scale [1]–[4], or initialized their models with a landmark of known size. Because orbiting spacecraft are in freefall, IMUs do not produce useful measurements of acceleration. For missions to unmapped destinations, like asteroids, it is desirable to produce measurements without prior knowledge of the orbited body's geometry.

In space, the state-of-practice is to use radio-based navigation. Radio produces measurements accurate to a few kilometers [5], but exhibits unacceptable latency at long range and relies on interaction with Earth. High latency is a challenge for highly precise operations near distant targets (i.e., asteroids and comets). Reliance on terrestrial infrastructure has significant detrimental effects on accuracy of scientific data when Earth is not in view. For example, because moon-orbit satellites rely on radio from Earth for localization in the process of modeling the moon's gravity, the gravity model of the far side of the moon is noticeably less accurate than that of the near side.

Our Monte Carlo simulation results show filter convergence of both spacecraft and landmark states. By incorporating a rough model of gravity in the filter process, scale is explicitly resolved.

Since no prior reference map is necessary, orbital SLAM enables true infrastructure-free spacecraft autonomy. It facilitates travel to unmapped bodies including asteroids and planets and improves low altitude flight, including landing, fly-by missions, and sample return, where maps at the necessary resolution do not exist. Furthermore, since a map is continuously being built and updated throughout navigation, orbital SLAM facilitates rapid modeling to scale and increases localization precision.

II. Related Work

In deep space, camera-based navigation measures angles to distant asteroids to triangulate location [6]–[8]. These observations rely on the asteroid appearance as a single pixel in an imager, reducing measurements to discriminating a point source from black background. Objects that appear larger than a few pixels introduce significant angular error. The Voyager missions used optical measurements with radio to Earth for navigation. Images of the planet's natural satellites against a starry background were used with knowledge of ephemerides to update a navigation filter run on Earth [6], [9].

Terrain-relative planetary orbit determination has demonstrated success in simulation [6]. This approach to visual navigation matches terrain appearance to pre-existing surface imagery and uses orbital dynamics to determine spacecraft position, velocity, and trajectory orbital parameters. Though successful in simulation, this method is achieved as a batch process and assumes an accurate existing map of the orbited body.

Additionally, SLAM has been studied for years as its own problem, beginning with Smith, Self, and Cheeseman's 1990 text [10]. Sebastian Thrun's *Probabilistic Robotics* text includes a detailed review of the state of art in indoor SLAM techniques [11]. Visual SLAM has been demonstrated in several applications [12], including Bearings-Only SLAM [13] and monocular SLAM [14], however, vision alone cannot measure scale. Furthermore, previous techniques use inertial data to resolve for scale in SLAM [1]–[4], however, it is important to note that an IMU will not work in orbit. Anything in orbit is in free fall and an IMU is a specific force sensor and is therefore unable to provide any useful measurements of acceleration due to gravity. The distinction of this work is that it explicitly models gravity in orbit to resolve for scale in the monocular SLAM problem. We demonstrate that this technique produces stable results through a series of Monte Carlo experiments.

III. Method

A. Overview

Orbital SLAM is implemented as an Extended Kalman filter [15]. The filter tracks the vehicle position and velocity as well as the state of the orbited body using a co-state representation. Visually distinct points on the body are tracked and stored as 3D "landmarks" representing the shape of the orbited body. The spacecraft state and surface landmarks are initialized to their true values with large amounts of added error and then are propagated forward in time in the process update using knowledge of orbital mechanics. Measurements are obtained by tracking the surface landmarks in camera images of the orbited body over time. The filter processes them in the measurement update as bearing angles, updates spacecraft position and velocity, and corrects landmark locations over time, thus building and continuously correcting a map of the orbited body. The goal is to show the covariance of the landmark locations and the spacecraft state decrease over time to show that the orbital gravity model stabilizes the estimate. A simplyifing assumption of this work is that the planet is not moving for the purposes of detecting and tracking features. This is discussed in more detail as future work.

B. Mathematical Derivation of Orbital SLAM

1) Process Update: The state vector contains information about both the spacecraft and its surroundings. In the case of orbital SLAM, spacecraft position and velocity along with landmark locations are tracked in three dimensions. Landmarks are considered to be surface features on the orbited body. Specifically, the spacecraft state vector, $\mathbf{x_s}$ (Eq. 1) along with the estimated map of landmark locations $\mathbf{x_l}$ (Eq. 2) make up the total combined state vector $\mathbf{x}$ as defined by Eq. 3.

$$\mathbf{x_s} = \begin{bmatrix} \mathbf{r_s} \\ \mathbf{v_s} \end{bmatrix} \begin{array}{l} \text{Spacecraft Position} \\ \text{Spacecraft Velocity} \end{array} \tag{1}$$

$$\mathbf{x_l} = \begin{bmatrix} \mathbf{r_{l_1}} \\ \vdots \\ \mathbf{r_{l_N}} \end{bmatrix} \quad \text{Position of } N \text{ landmarks} \tag{2}$$

$$\mathbf{x} = \begin{bmatrix} \mathbf{x_s} \\ \mathbf{x_l} \end{bmatrix} \tag{3}$$

The dynamics of the system are modeled using a spherical gravity model and assume no additional perturbations. It is also assumed surface landmarks do not move over time. Eq. 4 gives the continuous-time motion model of the state, where Eq. 5 defines the acceleration on the spacecraft due to the planet. In Eq. 5, G represents the gravitational constant, m_m represents the mass of the planet, and $\mathbf{r_{ms}}$ represents the vector from the planet to the spacecraft.

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{r}}_\mathbf{s} \\ \dot{\mathbf{v}}_\mathbf{s} \\ \dot{\mathbf{r}}_\mathbf{l_1} \\ \vdots \\ \dot{\mathbf{r}}_\mathbf{l_N} \end{bmatrix} = \begin{bmatrix} \mathbf{v_s} \\ \mathbf{a_s} \\ \mathbf{0}_{(3,1)} \\ \vdots \\ \mathbf{0}_{(3,1)} \end{bmatrix} \tag{4}$$

$$\mathbf{a_s} = \frac{-Gm_m\mathbf{r_{ms}}}{\|\mathbf{r_{ms}}\|^3} \tag{5}$$

The motion described by Eq. 4 can be discretized as in Eqs. 6 and 7. $\mathbf{M}$ is defined as the motion matrix and describes how the state changes with time.

$$\mathbf{x_t} = \mathbf{x_{t-1}} + \mathbf{M} \tag{6}$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{v_{s_{t-1}}}\Delta t + \frac{1}{2}\mathbf{a_{s_{t-1}}}\Delta t^2 \\ \mathbf{a_{s_{t-1}}}\Delta t \\ \mathbf{0_{(3N,1)}} \end{bmatrix} \tag{7}$$

The state transition matrix, $\mathbf{A}$, as defined by Eq. 8, is used to propagate the state $\mathbf{x}$ forward in time, therefore it contains the dynamics of both the spacecraft and landmarks. It can be partitioned into two pieces; one for propagating the spacecraft state, $\mathbf{A_s}$ (Eq. 9) and the identity matrix, $\mathbf{I}$, since it is assumed the landmarks are not moving over time. Specifically, $\mathbf{A_s}$ contains the Keplerian motion of the spacecraft in orbit noting that a_s is a scalar defined by Eq. 10.

$$\mathbf{A} = \begin{bmatrix} \mathbf{A_s} & \mathbf{0}_{(6,3N)} \\ \mathbf{0}_{(3N,6)} & \mathbf{I}_{3N} \end{bmatrix} \tag{8}$$

$$\mathbf{A_s} = \begin{bmatrix} \mathbf{I}_3 + (\frac{1}{2}a_s\Delta t^2)\mathbf{I}_3 & \Delta t\mathbf{I}_3 \\ (a_s\Delta t)\mathbf{I}_3 & \mathbf{I}_3 \end{bmatrix} \tag{9}$$

$$a_s = \frac{-Gm_m}{\|\mathbf{r_{ms}}\|^3} \tag{10}$$

The process update uses the known dynamics of the system to continuously propagate the state and covariance forward in time. Eq. 11 gives the state update and is a linearized form of Eq. 6. Eq. 12 gives the covariance update where the superscripts $-$ and $+$ indicate *a priori* and *a posteriori* a measurement update.

$$\mathbf{x_t} = \mathbf{A}\mathbf{x_{t-1}} \tag{11}$$

$$\mathbf{\Sigma_t^-} = \mathbf{G}\mathbf{\Sigma_{t-1}^+}\mathbf{G}^T + \mathbf{F}^T\mathbf{Q}\mathbf{F} \tag{12}$$

$\mathbf{Q}$ is a six- dimensional square matrix that represents the process noise covariance and $\mathbf{F}$ (Eq. 19) extends it to a square matrix of size $3N + 6$.

$$\mathbf{F} = \begin{bmatrix}\mathbf{I}_6 & \mathbf{0}_{(6,3N)}\end{bmatrix} \tag{13}$$

$\mathbf{G}$ is the derivative of the motion model given by Eq. 6. As a result of its additive form, it can be expanded to the form given by Eq. 14 where $\mathbf{g}$ is the Jacobian of motion matrix $\mathbf{M}$ with respect to the full state given by Eq. 15.

$$\mathbf{G} = \mathbf{I}_{3N+6} + \mathbf{F}^T\mathbf{g}\mathbf{F} \tag{14}$$

$$\mathbf{g} = \frac{\partial \mathbf{M}}{\partial \mathbf{x}} \tag{15}$$

2) Measurement Update: Measurements are represented as the angles-only spherical coordinate representation of a surface landmark in the reference frame of the spacecraft as shown in Fig. 1. Therefore, the measurement model is given by Eq. 16 where the subscripts l and s represent the landmark and spacecraft respectively.

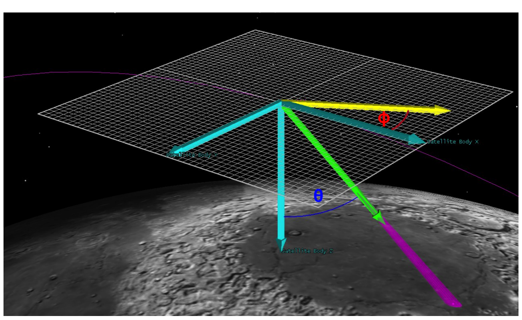


Figure 1. This shows a visualization of a sample measurement. The spacecraft's reference frame is indicated by the teal vectors, while its XY plane is indicated by the white grid. The green vector represents that to a landmark, while its projection in the satellite's XY plane is indicated by the yellow vector. The two angles, θ and ϕ, are the measurement received for each landmark.

$$\hat{\mathbf{z}} = \begin{bmatrix}\theta \\ \phi\end{bmatrix} = h(\mathbf{x}) = \begin{bmatrix}\arccos\left(\frac{z_l - z_s}{\sqrt{(x_l-x_s)^2+(y_l-y_s)^2+(z_l-z_s)^2}}\right) \\ \arctan\left(\frac{y_l-y_s}{x_l-x_s}\right)\end{bmatrix} \tag{16}$$

Measurements of landmarks are processed sequentially at a given time. Therefore, z_j represents a single measurement of the j-th landmark for all N landmarks. To calculate the Kalman gain, it is necessary to calculate the Jacobian of the measurement model, $\mathbf{H}$, with respect to the full state as given by Eq. 17.

$$\mathbf{H} = \mathbf{h}\mathbf{F_j} \tag{17}$$

Since measurements are processed sequentially, the Jacobian will only depend on the spacecraft location and the location of the j-th landmark. Therefore, $\mathbf{H}$ can be factored into a lower dimensional Jacobian $\mathbf{h}$, given by Eq. 18, and the matrix $\mathbf{F}$, which maps $\mathbf{h}$ into a matrix of the dimension of the full state vector [11].

$$\mathbf{h} = \left.\frac{\partial h(\mathbf{x})}{\partial \mathbf{x_s}, \mathbf{r_{l_j}}}\right|_{\mathbf{x}^-} \tag{18}$$

$$\mathbf{F_j} = \left[\begin{array}{c|c|c|c}\begin{matrix}\mathbf{I}_6 \\ \mathbf{0}_{(3,6)}\end{matrix} & \mathbf{0}_{(6+3,3j-3)} & \begin{matrix}\mathbf{0}_{(6,3)} \\ \mathbf{I}_3\end{matrix} & \mathbf{0}_{(6+3,3N-3j)}\end{array}\right] \tag{19}$$

The Kalman gain is then computed in Eq. 20, noting that $\mathbf{R}$ represents the measurement noise covariance, so that full state and covariance can be updated in Eqs. 21 and 22.

$$\mathbf{K} = \mathbf{\Sigma}\mathbf{H^T}[\mathbf{H}\mathbf{\Sigma}\mathbf{H^T} + \mathbf{R}]^{-1} \tag{20}$$

$$\mathbf{x_t} = \mathbf{x_{t-1}} + \mathbf{K} \cdot (\mathbf{z} - \hat{\mathbf{z}}) \tag{21}$$

$$\mathbf{\Sigma_t} = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{\Sigma_{t-1}} \tag{22}$$

IV. Simulated Landmark Detection

The SLAM approach was validated using a high fidelity visual simulation of the moon. A model produced by the NASA LRO team [16] was rendered using a raytracer [17] and lit using a sun lamp to produce highly realistic imagery representative of a lunar mission. The model is accurate to within 20 meters in the local horizontal plane and 1 meter in radius. It accounts for variation in the radius of the moon [18]. The raytracer models reflectance using the Oren-Nayer [19] reflectance model. This is a commonly used approximation of the true reflectance model of the moon.

At each filter measurement time (every 30 seconds), an image was produced using the renderer and processed as follows to simulate a measurement. The image was processed using Speeded up robust features (SURF) [20] to detect features in the current frame. Each SURF descriptor in the current frame was matched to a database of features to produce a feature identifier. The pixel coordinate of the feature in the image was converted into a bearing angle in the body frame of the spacecraft. The body frame measurement was converted to a world frame (J2000 [21]) coordinate by incorporating a simulated measurement from a star tracker. Representative noise for the star tracker was added to produce a measurement that accurately represents the process commonly utilized in spacecraft.

Images were rendered using a pinhole camera model with the following intrinsic parameters: focal length of 35mm, image sensor size of 32mm x 18mm, and image resolution of 1920 x 1080 pixels. The principal point was assumed to be at the center of the image sensor.

Because the focus of this paper is in evaluating the stability of tracking after initialization, features were initialized using a highly perturbed version of ground truth. Feature tracking was detected features and discarded ones with more than 20km of error from the matching feature.

The pipeline for simulating landmark detection is shown in Figure 2.

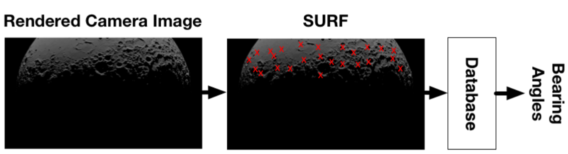


Figure 2. Flow chart of process from rendering image to generating bearing angle

V. Results

The orbital SLAM filter was tested using two techniques. First it was tested in Monte Carlo simulation to demonstrate filter convergence and to show the orbital gravity model resolves for scale. Then, it was tested in conjunction with a landmark detection and tracking algorithm on a high-fidelity moon model. This high-fidelity simulation is more realistic of the conditions that would be encountered in space and therefore more accurately models the errors in measurements that would be seen in a real mission.

A. *Monte Carlo Simulations*

Two Monte Carlo experiments were run to test Orbital SLAM. The goal is to show the covariance and error of the spacecraft state and landmark locations decreasing over time. Successful simulation demonstrates that knowledge of gravity in orbit resolves scale for orbital SLAM. A lunar orbit was generated in AGI's Satellite Toolkit (STK) with orbital parameters given in Table I.

Table I. Test Orbit Parameters

a	e	i	Ω	ω	M
2037.4 km	$1 \cdot 10^{-6}$	$0°$	$0°$	$0°$	$0°$

Landmark locations were randomly chosen for each simulation as uniformly distributed points on a sphere of radius 1737.4 km, representing an estimate of the average radius of the Moon. Spacecraft position and velocity as well as landmark locations were initialized to their true value with added normally distributed random noise. It was assumed that landmarks could be seen at all times, and the Moon was considered to be non-rotating. The first simulation had initial noise added to the spacecraft location on the order of 10 km, to the spacecraft velocity on the order of 0.1 km/s, and to the landmark locations on the order of 1 km. Simulated measurements had noise on the order of 0.01 radians added. The algorithm was run 25 times and 30 landmarks were used for 500 simulated minutes. Fig. 3 shows the results of this initial test, where the green orbit and the teal and black circles represent truth, and the red and magenta lines represent the estimated orbit and landmarks.

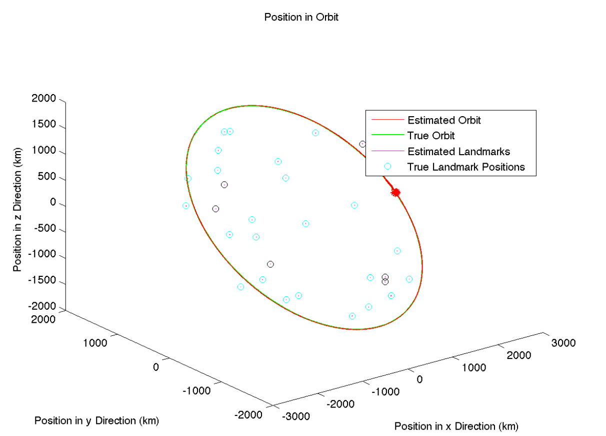
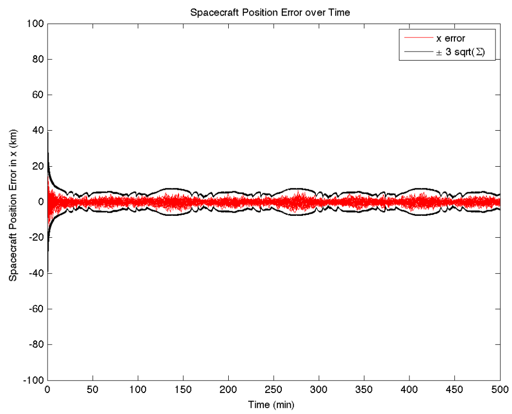


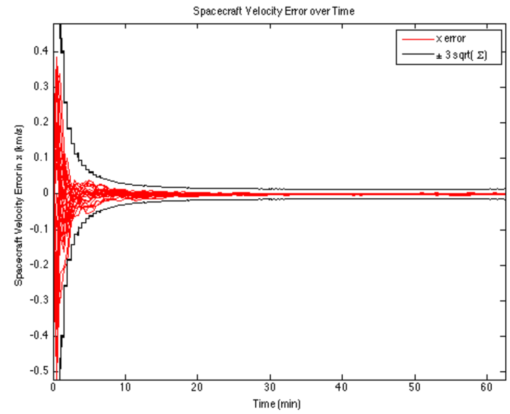
Figure 3. This plot shows the first Monte Carlo simulation results. Noise on the order of 10 km was added to initial spacecraft location, on the order of 0.1 km/s to initial spacecraft velocity, and on the order of 1 km to initial landmark locations. The results of 25 runs are plotted on top of truth orbit and landmarks, their close proximity indicating good convergence.

The red dot represents the starting location of the spacecraft in orbit. The position of the numerous red orbits on top of the green truth orbit and the placement of the magenta landmarks inside the teal and black truth landmarks are indications of good convergence. This is further validated by Figs. 4a, 4b, and 4c. Specifically, Fig. 4a shows spacecraft position error in the x dimension plotted over time and its corresponding covariance. Algorithm convergence is validated by position error for all 25 runs falling within $\pm 3\sqrt{\sigma}$ and by the error being centered at zero. Similarly, Fig. 4b shows spacecraft velocity error in the x dimension over time and its corresponding covariance zoomed in and Fig. 4c shows a sample landmark's position error in the x dimension and its corresponding covariance over time.

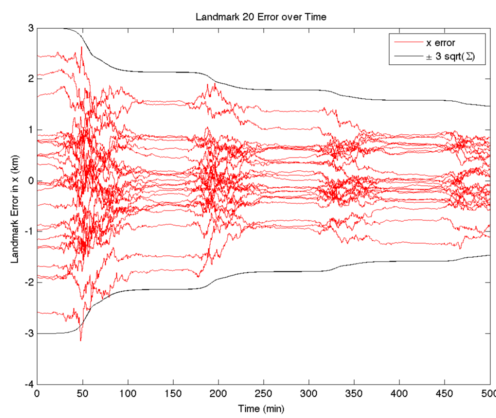
Since the algorithm converged with the amount of noise indicated, tests were run to see how large of an initialization



(a) Spacecraft position error in X and covariance plotted over time.

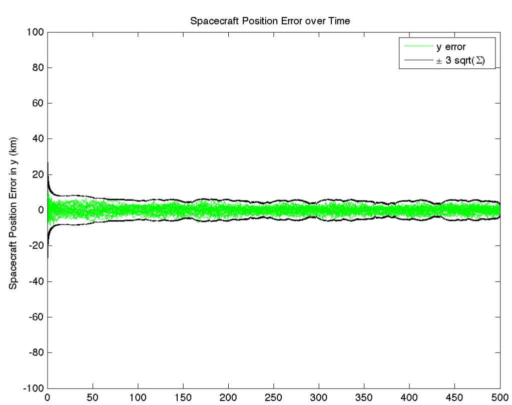


(b) Spacecraft velocity error in X and covariance plotted over time.

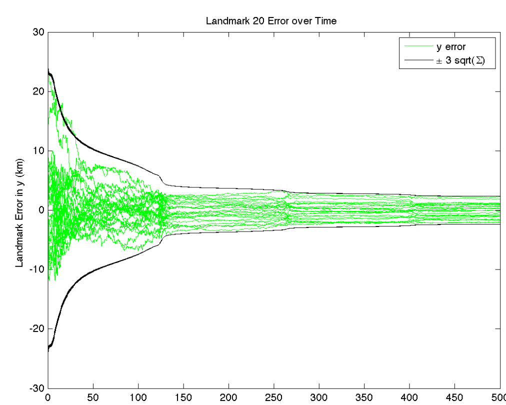


(c) Single landmark position error in X and covariance plotted over time.

Figure 4. Algorithm convergence is validated by the error falling within $\pm 3\sqrt{\sigma}$ for all runs of the Monte Carlo simulation.

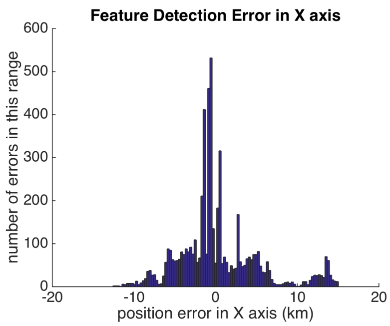


(a) Spacecraft position error in Y and covariance plotted over time with increased initialization error.

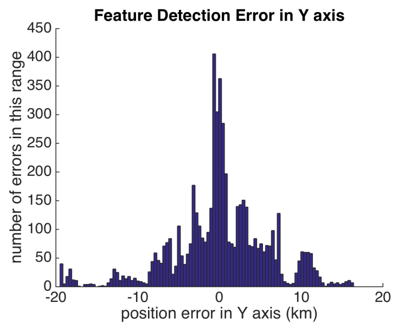


(b) Spacecraft velocity error in Y and covariance plotted over time with increased initialization error.

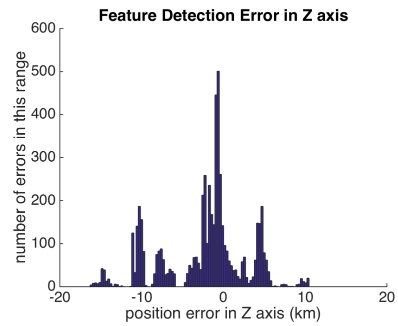
Figure 5. Algorithm convergence is validated by the error falling within $\pm 3\sqrt{\sigma}$ for all runs of the Monte Carlo simulation.



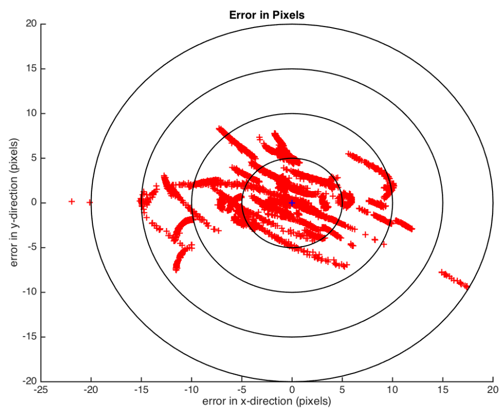
(a) Noise models in X of the same landmarks shown in Figure 6d.



(b) Noise models in X of the same landmarks shown in Figure 6d.



(c) Noise models in X of the same landmarks shown in Figure 6d.



(d) This figure illustrates the noise model for all tracked landmarks in the 1200 minute orbit by plotting the error in pixels.

Figure 6. Noise model of landmark tracking simulation

error it could handle. The second test was run with error on the order of 10 km in spacecraft position error, of 1 km/s in spacecraft velocity error, and of 10 km in landmark location error. Measurement noise was kept at 0.01 radians. Fig. 5a shows spacecraft position error over time, this time in the y dimension, while Fig. 5b shows landmark position error over time. These results show that the Monte Carlo runs stay within the filter covariance bounds, and that the covariance is decreasing, indicating convergence.

B. Landmark Tracking Simulation Results

The parameters of the Clementine orbit used are given in Table II where θ represents the true anomaly.

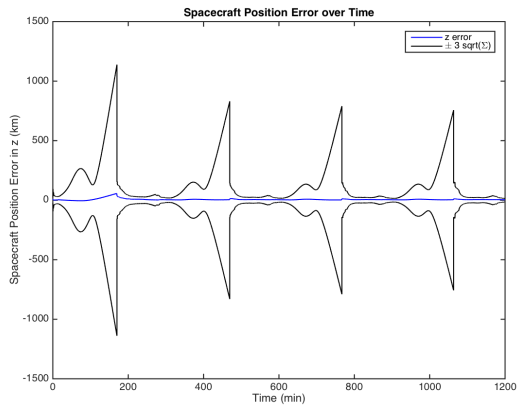
Table II. Clementine Orbit Parameters

a	e	i	Ω	ω	θ
3414.5 km	0.3678	114.146°	3.201°	26.059°	162.997°

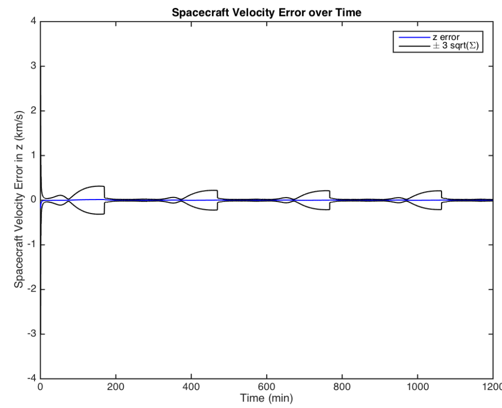
Landmarks were tracked through 1200 minutes worth of images and measurements were received by the filter every 30 simulated seconds. A standard deviation of 10 km initialization error was added to the landmarks and the spacecraft position. A standard deviation of 0.10 km/s initialization error was also added to the spacecraft velocity. The noise model for the landmark tracking is shown in Figure 6, with maximum error of 20 km. Error was capped by comparing to ground truth positions and removing landmarks from consideration if their distance from ground truth was more than x km (in this case x = 20). Figure 7 illustrates the spacecraft position, velocity, and a single landmark's position error plotted over time. The "steps" in the landmark position error (Fig. 7c) occur when the filter does not receive landmark measurements. This is a result of the inability to track landmarks when the spacecraft is in total darkness or when it is at periapsis and the camera's field of view becomes too narrow.

VI. Conclusions & Future Work

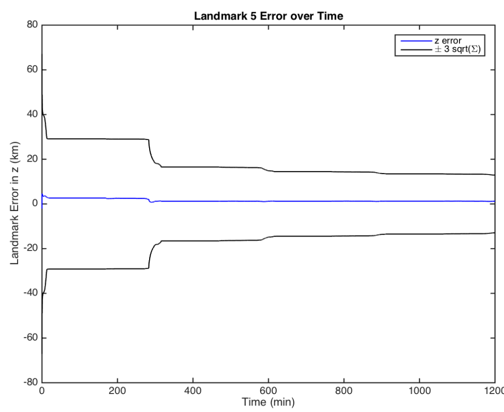
This paper has presented a method for orbital SLAM by tracking surface landmarks through camera images over time. The technique determines spacecraft position and velocity along with landmark locations using bearings-only measurements of landmarks and knowledge of orbital mechanics.This method enables true spacecraft autonomy by replacing radio localization with onboard localization. When combined with a gravity model, results show visual SLAM



(a) Spacecraft position error in Z and covariance plotted over time.



(b) Spacecraft velocity error in Z and covariance plotted over time.



(c) Single landmark position error in Z and covariance plotted over time.

Figure 7. Algorithm convergence is validated by the error falling within $\pm 3\sqrt{\sigma}$.

converges to the correct scale.

Future work lies in determining a proper method for landmark initialization. Currently, landmarks are initialized to their true location with large added error. More robust initialization methods exist in the literature [13] and future iterations of orbital SLAM will take advantage of these methods. In addition, further work should address field of view and the duration of landmark visibility, as well as take into account planetary rotation.

ACKNOWLEDGMENT

REFERENCES

[1] E. S. Jones and S. Soatto, "Visual?inertial navigation, mapping and localization: A scalable real-time causal approach," *The International Journal of Robotics Research*, vol. 30, no. 4, pp. 407–430, April 2011.

[2] J. Kelly, S. Saripalli, and G. S. Sukhatme, "Combined visual and inertial navigation for an unmanned aerial vehicle," in *Field and Service Robotics*, ser. Springer Tracts in Advanced Robotics, C. Laugier and R. Siegwart, Eds. Springer Berlin Heidelberg, 2008, vol. 42, pp. 255–264.

[3] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Robotics and Automation, 2007 IEEE International Conference on*, April 2007, pp. 3565–3572.

[4] N. Trawny, A. I. Mourikis, S. I. Roumeliotis, A. E. Johnson, J. F. Montgomery, A. Ansar, and L. H. Matthies, "Coupled vision and inertial navigation for pin-point landing," in *NASA Science and Technology Conference*, 2007.

[5] K. Galal, T. Colaprete, S. Cooley, and B. Kennedy, "Trajectory design and orbit determination for the lunar crater observation and sensing satellite (lcross)."

[6] K. Peterson, H. Jones, C. Vassallo, A. Welkie, and W. Whittaker, "Terrain-relative planetary orbit determination," in *International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, Turin, Italy, Sept. 4-6 2012.

[7] J. Riedel, S. Bhaskaran, S. Synnot, S. Desai, W. Bollman, P. Dumont, C. Halsell, D. Han, B. Kennedy, G. Null, J. Owen, W.M., R. Werner, and B. Williams, "Navigation for the new millennium: Autonomous navigation for deep- space 1," *Proc. Int. Symposium Space Flight Dynamics*, 1997.

[8] S. Bhaskaran, J. E. Riedel, B. Kennedy, T. Wang, and R. Werner, "Navigation of the deep space 1 spacecraft at borrelly," NASA, Tech. Rep., 2002.

[9] J. K. Campbell, S. P. Synnott, and G. J. Bierman, "Voyager orbit determination at jupiter," *IEEE Transactions on Automatic Control*, vol. AC-28, no. 3, 1983.

[10] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous Robot Vehicles*, I. Cox and G. Wilfong, Eds. Springer-Verlag, 1990, pp. 167 – 193.

[11] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, 1st ed. MIT Press, 2006.

[12] A. J. Davidson, "Real-time simultaneous localisation and mapping with a single camera," in *In Internation Conference on Computer Vision*, 2003, pp. 1403–1410.

[13] M. Deans, "Bearings-only localization and mapping," Ph.D. dissertation, tech. report, CMU-RI-TR-05-41, Robotics Institute, Carnegie Mellon University, 2005.

[14] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Trans Pattern Anal Mach Intell.*, vol. 29, no. 6, pp. 1052 – 1067, June 2007.

[15] G. A. Einicke and L. White, "Robust extended kalman filter," *IEEE Trans. Signal Processing*, vol. 47, no. 9, pp. 2596–2599, 1999.

[16] N. Petro and J. Keller, "Five years at the moon with the lunar reconnaissance orbiter (lro): New views of the lunar surface and environment," *Annual Meeting of the Lunar Exploration Analysis Group*, 2014.

[17] B. Foundation, "Blender 2.69," http://www.blender.org/download/.

[18] E. Mazarico, D. Rowlands, G. Neumann, D. Smith, M. Torrence, F. Lemoine, and M. Zuber, "Orbit determination of the lunar reconnaissance orbiter," *Journal of Geodesy*, vol. 86, no. 3, pp. 193–207, 2012.

[19] M. Oren and S. Nayer, "Generalization of lambert's reflectance model," *SIGGRAPH*, pp. 239–246, 1994.

[20] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *In ECCV*, 2006, pp. 404–417.

[21] S. Tapley and Born, *Statistical Orbit Determination*. Elsevier Academic Press, 2004, pp. 29–32.