SQUARE ROOT EXTENDED INFORMATION FILTER VISUAL ODOMETRY APPLIED TO BLUE ORIGIN DEORBIT, DESCENT, AND LANDING DATASET

Matthew W. Givens[1*] and Jay W. McMahon[1]; [1]Colorado Center for Astrodynamics Research, 429 UCB, 3772 Discovery Drive, Boulder, CO 80303, [*][matthew.givens@colorado.edu]

Abstract. *A novel square root extended information filter (SREIF) has been developed to address the visual-inertial odometry and SLAM problems with the potential for constant-time complexity navigation and mapping with complex dynamics and large maps. This algorithm will be applied to a recent, publicly-available dataset, the Blue Origin Deorbit Descent, and Landing Tipping Point Dataset, and the filter's performance will be assessed against the post-processed truth solution. Further, the necessary discrete-time IMU dynamics models, which are not as commonly accessible as their continuous-time counterparts, will be explicitly derived and provided for the benefit of future researchers and practicing engineers.*

Introduction. Terrain Relative Navigation (TRN) is traditionally characterized by methods that aim to match onboard sensor data to pre-existing onboard maps in order to improve a spacecraft's absolute position and velocity estimates relative to an absolute frame. This requires that an a priori map be available before flight, perhaps based on data gathered by a prior mapping spacecraft or the like, or an iterative process wherein ground personnel are involved in creating and exchanging maps with the spacecraft during flight. For destinations farther afield from Earth, beyond the moon and Mars, these prior maps are not currently available for most bodies and future autonomous and semi-autonomous missions will need to have the capability to extract navigation observables from potentially unmapped terrain expediently in order to make prompt guidance and control decisions. The most versatile frontend sensor for these scenarios is commonly considered to be the optical camera due to its ubiquity, small size and weight, and passivity.

The backend estimation problem of navigating through an unknown environment while contemporaneously mapping it is referred to in the robotics community as the Simultaneous Localization and Mapping (SLAM) problem. Starting with solutions based on the Extended Kalman Filter (EKF) in the 1980s, the state-of-the-art progressed to particle filtering and graph- and keyframe-based least squares methods more recently.[1] There is thus a deep and rigorous body of research documenting the development of these techniques that has shed light on the underlying information structure inherent to the SLAM estimation problem. Visual Odometry (VO) can be formulated in a very similar way to V-SLAM but, in favoring only the trajectory and not the map, tends to "forget" visual features over time in order to maintain computational efficiency.[2] This precludes the use of loop closures as in full SLAM which serve to improve the overall solution by the re-detection of previously tracked visual landmarks.

The EKF, while generally out of favor in the SLAM community due to its computational scalability, has a long and continuing history of successful application in aerospace problems and is well-suited to complex dynamics models and the inclusion of process noise. To the former point, where complex pre-integration strategies must be used in graph-based approaches to account for high-rate sources such as from inertial measurement unit (IMU) data, the EKF handles this efficiently through a "model replacement" kinematics model. Likewise, the inclusion of process noise, essential in space problems to account for unknown perturbations, is notoriously difficult in least-squares algorithms and relatively simple to include in the EKF.[3]

These observations motivate the creation of a new EKF-like SLAM algorithm for space applications that specifically targets the EKF's main drawback: its unbounded computational complexity. The applications of such a filter are imagined to be in a variety of spacecraft relative navigation scenarios including TRN for descent and landing, small body relative navigation, and proximity operations and rendezvous with an unknown spacecraft. Here, we briefly present the proposed solution and apply it to the recent Blue Origin Deorbit, Descent, and Landing Tipping Point Flight 1 dataset[4] which includes images, IMU data, LIDAR data, and a post-processed truth solution from a suborbital test launch of the New Shepard rocket.

Information Filtering. Kalman filters track a mean and associated Gaussian uncertainty in the form of a state vector and covariance matrix. The canonical form of the covariance matrix is its inverse, the so-called information matrix, which appears explicitly in the typical definition of the Gaussian probability density function. SLAM filters based on the information matrix have been devised previously by researchers to take advantage of the information form's inherent sparsity, additive measurement updates, and intuitive mapping to the Markov Random Field (MRF) graphical model.[5] These filters all suffered from drawbacks associated with approximations related to preserving sparsity and inverting the information matrix for state propagation.

Information filtering is not a foreign concept to the space navigator and algorithms based on the information square root are not uncommon in orbit determination. In fact, Bierman[6] emphatically argues that square root factorization methods are strictly superior to Kalman-type filters in every conceivable scenario due to their inherent numerical stability. Interestingly, this observation circa 1977 may yet hold true for the sequential SLAM problem.

Following insights from these earlier works, as well as the observation by Daellaert & Kaess[7] that the square root (SR) of the information matrix maps to a Bayes Tree instead of an MRF, we instead see an opportunity to formulate the visual SLAM problem using the SR information matrix. Our previous work has shown that this structure can be leveraged to create a tree-like dependence structure rooted at the current state estimate. This enables easy access to the exact marginal distributions required for state propagation as well as fast recovery of the states via simple back-substitution. Further, because this structure is created incrementally, it can be of arbitrary absolute size, memory permitting, without adversely affecting the baseline computational requirements of the algorithm.

Visual Features. Image features identified and tracked over multiple image frames, notoriously unreliable in their natural XYZ parameterization within sequential filtering algorithms, can instead be accurately captured using the Inverse Depth Points (IDP) measurement model.[8] This approach requires that a copy of the camera position for each landmark at the time it is instantiated to "anchor" it to the world reference frame. By over-parameterizing each landmark in terms of this position and some world-referenced spherical angles, as well as the inverse of the initial range, the tracking performance is much more linear and therefore Gaussian over multiple images. The downside is that the state dimension is larger than it would be otherwise, $3p + 3n \times 1$ instead of $3n$ where p is the number of anchor poses and n is the number of landmarks in the state. Our targeted handling of the computational complexity, as well as our approach of incrementally augmenting states to preserve sparsity, naturally addresses this drawback.

Dynamics Model. The current literature in the realm of visual-inertial navigation systems is awash with IMU model replacement formulations based on the assumption that the IMU output data to the filter is in the form of raw specific force and angular velocity. By contrast, many real higher-quality IMUs provide a lower-rate data output (50-100 Hz) that is pre-integrated in the form of delta velocity Δv and delta angle $\Delta\theta$ measurements that are valid over the sampling interval and account for coning and sculling effects. Such is the case with the BO DDLTP dataset, where these quantities are given at 50 Hz. One approach may be to convert these quantities back to average specific forces and angular rates via a simple division by the intervening time intervals but we prefer a closed-form solution based on these raw integrated data values.

The state vector we have chosen to use includes position and velocity in the Earth-Centered-Earth-Fixed (ECEF) frame E as well as a quaternion to represent the attitude from the Center of Navigation (CON) frame N,

$$\mathbf{x} = \begin{bmatrix} {}^E\mathbf{p} & {}^E\mathbf{v} & {}^E_N q \end{bmatrix}^T .$$

These were processed in a "model replacement" mode where the nominal trajectory in the state propagation comes from the high-rate IMU outputs. The full position and velocity states can be updated over $\Delta t = t_{k+1} - t_k$ via

$$ {}^E\mathbf{p}_{k+1} = {}^E\mathbf{p}_k + \Delta t {}^E\mathbf{v}_k + \frac{\Delta t}{2} {}^E\Delta\mathbf{v}_c \tag{1}$$

$$ {}^E\mathbf{v}_{k+1} = {}^E\mathbf{v}_k + {}^E\Delta\mathbf{v}_c \tag{2}$$

where the true accelerometer output comes into the equations through

$$ {}^E\Delta\mathbf{v}_c = {}^E_B\mathbf{C} {}^B\Delta\mathbf{v} + {}^E\Delta\mathbf{v}_g $$

and where ${}^E_B\mathbf{C}$ represents the rotation matrix mapping vectors from the IMU frame B to the ECEF frame. The gravitational and Coriolis effects must be accounted for via

$$ {}^E\Delta\mathbf{v}_g = \Delta t({}^E\mathbf{g} - 2{}^E\boldsymbol{\omega}_\oplus \times \mathbf{v}_k) $$

where ${}^E\boldsymbol{\omega}_\oplus$ is the rotation rate of the Earth in radians per second and ${}^E\mathbf{g}$ is the acceleration due to gravity. Since the IMU itself is at a lever arm from the CON, the difference in specific forces at that location must be accounted for. Referring to Steffes,[9] this can be done via

$$ {}^B\Delta\mathbf{v} = {}^B\Delta\mathbf{v} - {}^B\Delta\mathbf{v}_{rot} $$

with

$$ {}^B\Delta\mathbf{v}_{rot} = {}^B\Delta\mathbf{v}_{cent} + {}^B\Delta\mathbf{v}_{Euler} $$

$$ {}^B\Delta\mathbf{v}_{cent} \approx -\frac{1}{\Delta t}([{}^B\Delta\boldsymbol{\theta}\times]^{\circ 2} \circ {}^B\mathbf{d})[\mathbf{1}] $$

$$ {}^B\Delta\mathbf{v}_{Euler} \approx \mathbf{0} $$

where $\circ$ represents an elementwise (Hadamard) square, ${}^B\mathbf{d}$ is the lever arm in the IMU frame, and $[\mathbf{1}]$ is a column of ones. The delta velocity output from the IMU is corrupted by zero-mean, white Gaussian noise,

$$ {}^B\Delta\tilde{\mathbf{v}} = {}^B\Delta\mathbf{v} + {}^B\mathbf{w}_v $$

Defining the additive error states as

$$ {}^E\delta\mathbf{p} = {}^E\mathbf{p} - {}^E\hat{\mathbf{p}} $$

$$ {}^E\delta\mathbf{v} = {}^E\mathbf{v} - {}^E\hat{\mathbf{v}} $$

plugging in the above relationships, and ignoring second-order terms eventually results in the following error state models for position and velocity:

$$ {}^E\delta\mathbf{p}_{k+1} = {}^E\delta\mathbf{p}_k + \Delta t {}^E\delta\mathbf{v}_k + \frac{\Delta t}{2}({}^B\delta\mathbf{v}_{k+1} - {}^B\delta\mathbf{v}_k) \tag{3}$$

$$ \begin{aligned} {}^E\delta\mathbf{v}_{k+1} = (\mathbf{I}_3 - 2\Delta t[{}^E\boldsymbol{\omega}_\oplus\times]) {}^E\delta\mathbf{v}_k - \\ {}^E_{\hat{N}}\mathbf{C}[{}^N_B\mathbf{C}({}^B\Delta\tilde{\mathbf{v}} - {}^B\Delta\hat{\mathbf{v}}_{rot})\times]^N\delta\boldsymbol{\psi} - {}^E_{\hat{B}}\mathbf{C}^B\mathbf{w}_v. \end{aligned} \tag{4}$$

where the approximation ${}^E_B\mathbf{C} \approx {}^E_{\hat{N}}\mathbf{C}(\mathbf{I}_3 + [{}^N\delta\boldsymbol{\psi}\times]) {}^N_B\mathbf{C}$ was used and where $[{}^N\delta\boldsymbol{\psi}\times]$ represents the skew-symmetric cross product matrix of a set of error angles ${}^N\delta\boldsymbol{\psi}$ between the estimated and true Center of Navigation frame N. We also discount any effects that the gyroscope bias and noise have on the lever arm term as these are significantly nonlinear but small.

Using a multiplicative formulation to properly capture operations on $SO(3)$, a set of quaternion states can be updated by computing the error quaternion

$$ {}^{N}_{\hat{N}}\delta\hat{q} \approx \begin{bmatrix} {}^{N}\delta\boldsymbol{\psi}/2 \\ 1 \end{bmatrix} $$

$$ {}^{E}_{N}\hat{q}_{k+1} = {}^{E}_{\hat{N}}\hat{q}_{k} \otimes {}^{N}_{\hat{N}}\delta\hat{q}^{-1} \tag{5} $$

Note that for this work, in concert with the conventions used in the dataset, we utilize the JPL quaternion convention as described by Trawny & Roumeliotis.[10]

Since the angular rates are already integrated and a principal rotation vector is being provided by the IMU, the error state for the angular rates can be represented simply as

$$ {}^{N}\delta\boldsymbol{\psi} = {}^{N}_{B}\mathbf{C}^{B}\Delta\boldsymbol{\theta} $$

where the true delta angle is corrupted by a gyroscope bias and noise,

$$ {}^{B}\Delta\tilde{\boldsymbol{\theta}} = {}^{B}\Delta\boldsymbol{\theta} + {}^{B}\mathbf{w}_{\theta}. $$

such that

$$ {}^{N}\delta\boldsymbol{\psi}_{k+1} = {}^{N}\delta\boldsymbol{\psi}_{k} + {}^{N}_{B}\mathbf{C}^{B}\Delta\tilde{\boldsymbol{\theta}} - {}^{N}_{B}\mathbf{C}^{B}\mathbf{w}_{\theta}. \tag{6} $$

The covariance matrix can now be propagated by

$$ \mathbf{P}_{k+1} = \boldsymbol{\Phi}_{k+1,k}\mathbf{P}_{k}\boldsymbol{\Phi}^{T}_{k+1,k} + \boldsymbol{\Gamma}_{k+1,k}\mathbf{Q}_{k}\boldsymbol{\Gamma}^{T}_{k+1,k} $$

where the state transition matrices can be directly extracted from equations 3, 4, and 6 and

$$ \mathbf{Q}_{k} = \text{diag}\{\sigma_{v}^{2}\mathbf{I}_{3}, \sigma_{\theta}^{2}\mathbf{I}_{3}\} $$

Without any measurements or bias compensation, the altitude over time as a result of the presented state propagation strategy and using the present dataset is shown in Figure 2, showing good agreement. Further inspection, however, reveals that there is a significant unmodeled perturbation after launch that occurs prior to the first image in the dataset at $t = 25$ seconds, ultimately resulting in a landing position that is nearly 4.5 km from the truth. Because of this, we chose to start our VIO algorithm at $t = 25$ seconds.

SREIF Mechanization. The reader is encouraged to review our previously-published works on our SREIF algorithm[11] but there are some new details used here that merit discussion. Since the lever arm between the camera frame C and CON frame N is nearly 1.5 meters, a modification must be made to the inverse depth measurement model to increase its consistency. In particular, whereas the inverse depth model in the literature only depends on a given feature's states and the camera's position at initialization, the addition of the lever arm results in a new dependence on the camera's attitude at initialization.

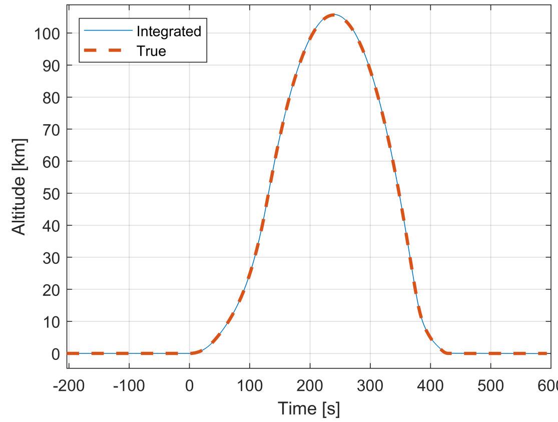

Figure 1. Altitude profile of integrated trajectory versus provided truth solution.

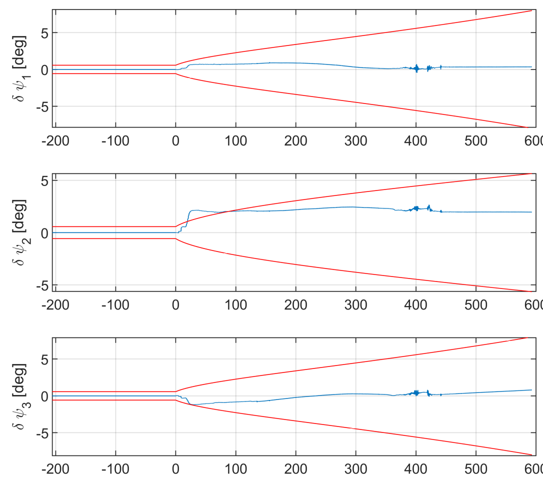

Figure 2. Attitude angles and $\pm 3\sigma$ over time. Note the unexplained temporary bias at liftoff, particularly in $\delta\psi_2$.

A measurement to a camera feature follows the pinhole camera model

$$ {}^{C}\mathbf{z} = \mathbf{K}\frac{{}^{C}\mathbf{h}}{{}^{C}\mathbf{h}_{z}} + \mathbf{w}_{m} $$

where $\mathbf{K}$ is the camera intrinsic matrix, $\mathbf{w}$ is the assumed zero-mean white Gaussian additive noise (in pixels), and the inverse depth model can be used to form the expected measurement

$$ {}^{C}\mathbf{h} = {}^{C}_{E}\mathbf{C}[\rho({}^{E}\mathbf{p}_{0} - {}^{E}\mathbf{p}) + {}^{E}\mathbf{m}] $$

where ρ is the inverse of the range to the feature from the initial camera position ${}^{E}\mathbf{p}_{0}$ and ${}^{E}\mathbf{m}$ is a vector of trigonometric functions that depends on globally-referenced azimuth θ and elevation ϕ angles. Details on these terms can be found in our recent works or in older inverse depth literature.[8] Here, we add in the additional complication of the camera lever arm ${}^N\mathbf{d}_C$ from the CON in the CON frame so that the inverse depth model becomes

$$ {}^C\mathbf{h} = {}^C_E\mathbf{C}[\rho({}^E\mathbf{p}_{C,0} - {}^E\mathbf{p}_C) + {}^E\mathbf{m}] $$

where the camera positions in the ECEF frame are expanded as

$$ {}^E\mathbf{p}_{C,0} = {}^E\mathbf{p}_0 + {}^E_N\mathbf{C}_0 {}^N\mathbf{d}_C $$
$$ {}^E\mathbf{p}_C = {}^E\mathbf{p} + {}^E_N\mathbf{C} {}^N\mathbf{d}_C $$

so that the measurement model, after simplification, becomes

$$ {}^C\mathbf{h} = \rho[{}^C_E\mathbf{C}({}^E\mathbf{p}_0 - {}^E\mathbf{p} + {}^E_N\mathbf{C}_0 {}^N\mathbf{d}_C) - {}^N\mathbf{d}_C] + {}^E\mathbf{m}] $$

which clearly has a dependence on the initial camera position and attitude as well as the current camera position and attitude and the inverse depth parameters. The relevant Jacobians are omitted here for brevity. In addition, measurement update iteration (as in Trawny et al[12]) was utilized to improve the convergence of the solution.

To identify features and tracks over the 350 useable images in the dataset, we utilize MATLAB's implementation of KAZE features and camera undistortion tools available in the Computer Vision Toolbox.[1*] A Mahalanobis distance test was also implemented for feature update outlier rejection.

Visual Odometry Results. After much effort in tuning and debugging, the presented VIO filtering strategy only marginal improvement on the raw IMU state propagation was achieved. Using realistic noise values and tracking as many features as were available, we observed that there is a strong bias in one of the attitude states that results in cascading errors in the other states. This also manifests in erroneous negative depths in the feature estimates. This can be seen in Figure 3 If the noise was increased to artificially de-weight the current estimate or the measurements, the resulting solution was noticeably improved and the bias attenuated. It is not currently known whether this was caused by an error in the formulation or with an unidentified programming bug.

Noteworthy is that by only tracking a single feature at a given time, the consistency of the solution can be improved and the previously-noted attitude bias is absent. Figure 5 shows the true trajectory and VIO solution algongside individually tracked landmarks on the ground. With this observation in mind, a likely culprit for the observed inconsistency may be the fact that using a single anchor camera pose for multiple features, as opposed to an anchor pose for each feature, can result in divergent behavior under certain conditions. This fact, as noted and improved upon in the literature,[13] was not known to the authors until very recently.

These results might also be improved by explicitly modeling and estimating IMU biases or other parameters during flight. Efficiently doing so can incur a computational

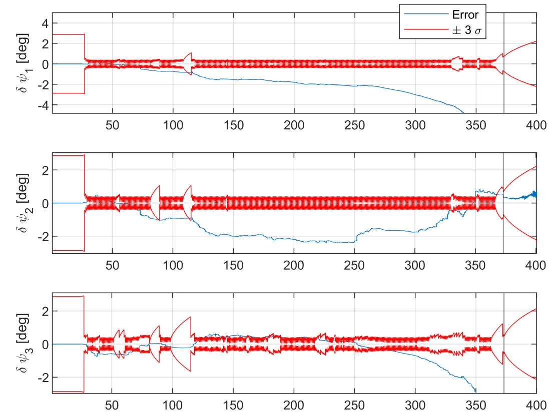

Figure 3. Attitude angle errors and $\pm 3\sigma$ *over time generated by the VIO algorithm. A vertical black line denotes the time of the last useable image. Note, in particular, the unexplained bias in* $\delta\psi_1$.

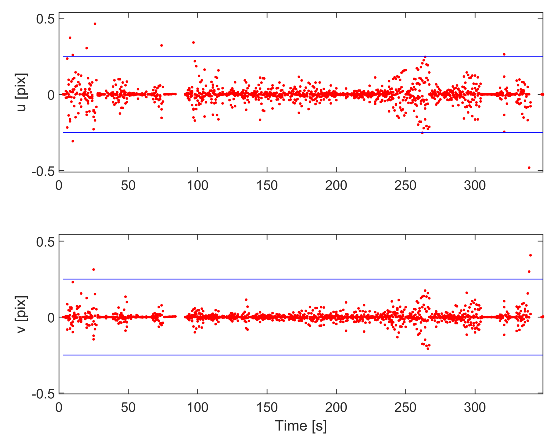

Figure 4. Post-fit pixel residuals with $\pm 1\sigma$ *when tracking many features with update iteration.*

penalty because of the state augmentation strategy employed in our SREIF and thus it is the subject of future research by the authors. Additionally, since the present scenario offers no opportunity for loop closures, as may or may not be the case in other relevant space navigation scenarios, the effects of loop closures on the computational performance are not relevant here and the solution is equivalent to a visual odometry approach but where the estimated map can be retained at no additional computational cost. Efficient loop closures and parameter estimation, as well as real-time implementation, will be the subject of future work.

Conclusion. While the filtering strategy presented was not entirely successful in improving the inertially-integrated IMU solution, we believe that the mathematics presented here will be useful to other researchers in the field that wish to work with the Blue Origin Deorbit

[1*]https://www.mathworks.com/products/computer-vision.html

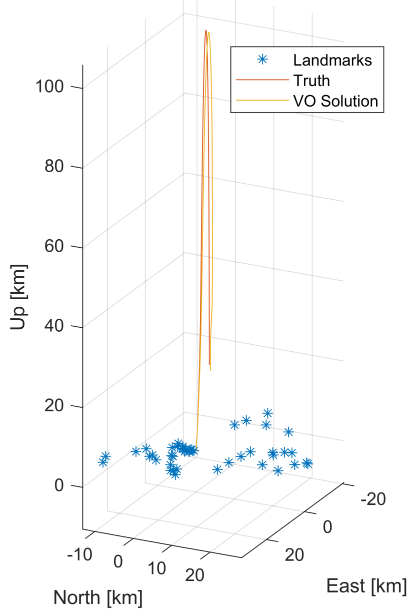

Figure 5. Comparison of true trajectory with VIO solution with only one feature tracked at a time.

Descent, and Landing Tipping Point Dataset. Since the discrete form of the dynamics shown here does not seem to be as commonly expressed in the literature as the continuous form, particularly when including corrections for the IMU lever arm, and since we believe that stating these equations explicitly will enable other researchers to more easily process this particular dataset, we have chosen to provide them in this paper.

References.

[1] H. Strasdat, J. M. Montiel, and A. J. Davison, “Visual SLAM: Why filter?,” *Image and Vision Computing*, vol. 30, pp. 65–77, feb 2012.

[2] J. A. Christian, L. Hong, P. McKee, R. Christensen, and T. P. Crain, “Image-based lunar terrain relative navigation without a map: Measurements,” *Journal of Spacecraft and Rockets*, vol. 58, pp. 164–181, oct 2021.

[3] B. D. Tapley, B. E. Schutz, and G. H. Born, *Statistical Orbit Determination.* Elsevier Inc., may 2004.

[4] S. Bieniawski, B. Lewis, B. Friia, A. Mahajan, K. Somervill, P. Mamidipudi, and D. Dakin, “New Shepard Flight Test Results from Blue Origin De-Orbit Descent and Landing Tipping Point,” *AIAA Science and Technology Forum and Exposition, AIAA SciTech Forum 2022*, 2022.

[5] S. Thrun, Y. Liu, D. Koller, A. Y. Ng, Z. Ghahramani, and H. Durrant-Whyte, “Simultaneous localization and mapping with sparse extended information filters,” *International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 693–716, 2004.

[6] G. J. Bierman, *Factorization methods for discrete sequential estimation.* Academic Press, 1977.

[7] F. Dellaert and M. Kaess, “Factor Graphs for Robot Perception,” *Foundations and Trends in Robotics*, vol. 6, no. 1-2, pp. 1–139, 2017.

[8] J. M. Montiel, J. Civera, and A. J. Davison, “Unified inverse depth parametrization for monocular SLAM,” *Robotics: Science and Systems*, vol. 2, pp. 81–88, 2007.

[9] S. R. Steffes, *Development and Analysis of SHEFEX-2 Hybrid Navigation System Experiment.* PhD thesis, 2013.

[10] N. Trawny and S. I. Roumeliotis, “Indirect Kalman Filter for 3D Attitude Estimation,” Tech. Rep. 2005-002, University of Minnesota, Minneapolis, MN, 2005.

[11] M. W. Givens and J. W. Mcmahon, “Square Root, Sequential Visual Odometry for Constant-Time Navigation and Mapping,” in *AIAA SCITECH 2022 Forum*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, jan 2022.

[12] N. Trawny, A. Mourikis, S. I. Roumeliotis, A. E. Johnson, and J. F. Montgomery, “Vision-Aided Inertial Navigation for Pin-Point Landing using Observations of Mapped Landmarks,” *Journal of Field Robotics*, vol. 24, no. 5, pp. 1–17, 2007.

[13] T. Pietzsch, “Efficient feature parameterisation for visual SLAM using inverse depth bundles,” in *BMVC 2008 - Proceedings of the British Machine Vision Conference 2008*, 2008.