

Extended Navigation Capabilities for a Future Mars Science Helicopter Concept

Jeff Delaune, Roland Brockers, David S. Bayard, Harel Dor, Robert Hewitt, Jacek Sawoniewicz,
Gerik Kubiak, Theodore Tzanetos, Larry Matthies and J. (Bob) Balaram

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Dr.
Pasadena, CA 91109

{jeff.h.delaune, roland.brockers, david.s.bayard, harel.dor, robert.a.hewitt, jacek.m.sawoniewicz,
gerik.kubiak, theodore.tzanetos, lhm, j.balaram}@jpl.nasa.gov

Abstract—This paper introduces an autonomous navigation system suitable for supporting a future Mars Science Helicopter concept. This mission concept requires low-drift localization to reach science targets far apart from each other on the surface of Mars. Our modular state estimator achieves this through range, solar and Visual-Inertial Odometry (VIO). We propose a novel range update model to constrain visual-inertial scale drift using a single-point static laser range finder, that is designed to work over unknown terrain topography. We also develop a sun sensor measurement model to constrain VIO yaw drift. Solar VIO performance is evaluated in a simulation environment in a Monte Carlo analysis. Range-VIO is demonstrated in flight in real time on 1 core of a Qualcomm Snapdragon 820 processor, which is the successor of the NASA's Mars Helicopter flight processor.

TABLE OF CONTENTS

1. INTRODUCTION.....	1
2. RELATED WORK	2
3. RANGE AND SOLAR VISUAL-INERTIAL ODOMETRY	2
4. TESTING	5
5. RESULTS	6
6. SUMMARY	8
ACKNOWLEDGMENTS	8
REFERENCES	8
BIOGRAPHY	9

1. INTRODUCTION

NASA's Mars 2020 mission will carry a Mars Helicopter (MH2020) to validate helicopter flight is possible in the ultra-thin Martian atmosphere [1] [2]. A successful demonstration would provide validated technologies and models necessary for the design of future helicopters for Mars exploration. The Mars Science Helicopter (MSH) concept envisioned in this paper is one such future helicopter concept. The objective of the MSH is to carry a moderate to large science payload (e.g. 2 kg of instruments) to designated surface locations in areas otherwise inaccessible to rovers. Such a helicopter could work in a standalone manner communicating to Earth via an orbiter relay or it could work cooperatively with a lander or rover

Both MH2020 and MSH require full reliable autonomy with no human intervention. This is necessitated by the long

time-delays for communication to Mars and the highly dynamic and uncertain flight environment. MSH will extend the MH2020 capabilities in real-time onboard navigation to estimate position, attitude and velocity in flight over rough, highly-sloped and even discontinuous terrain. To enable this, the global planar ground assumption of the MH2020 navigation algorithm [3] will be relaxed to allow the science helicopter to operate over such challenging terrains.

For robustness and accuracy, the state vector maintained in the on-board estimates is required to provide low-drifting state estimates under all expected motion conditions, including uniform-velocity translations which are not observable by visual-inertial state estimators typically used in robotics research [4]. That is a challenge because sensor options for MSH are extremely limited. Size, Weight, and Power (SWaP) requirements imposed by flying in an extremely thin atmosphere with limited battery capacity, prevent the use of most active sensors. The small rotorcraft dimensions introduce range limitations to stereo camera setups. Daytime outdoor flight conditions prevent the use of time-of-flight cameras based on the projection of an infrared pattern on the scene, which would be flooded by the Sun's radiation. Unlike sensors typically used on terrestrial Unmanned Air Vehicles (UAVs), Mars atmosphere is too thin for pressure sensors, and the magnetic field too weak for magnetometers. Finally, GPS cannot be used because it does not exist on Mars. For these reasons, we propose range and solar-visual-inertial odometry algorithms that provide the required low-drift guarantees using hardware comprised of a monocular camera, a static single-point Laser Range Finder (LRF), a sun sensor, and an Inertial Measurement Unit (IMU), that together meet the tight SWaP requirements of MSH.

This paper claims the following contributions:

- a low-drift state estimator able to process range, solar, visual and inertial measurements.
- a range measurement model for unknown terrain topography, which limits the visual-inertial scale drift in the absence of IMU excitation.
- evaluation of range-visual-inertial navigation performance on real data
- real-time demonstration with spaceflight-representative hardware
- a sun sensor measurement model, which limits the visual-inertial yaw drift.
- evaluation of solar-visual-inertial navigation performance on simulated data

Section 2 reviews visual-inertial odometry in the robotics research literature and the existing solutions to limit the

associated error drift, using range and sun sensors. Section 3 presents our state estimation framework based on an extended Kalman filter. This includes state-of-the-art visual-inertial Simultaneous Localization And Mapping (SLAM). Our key contribution is a range measurement update that is designed to observe the scale, including during the acceleration-free parts of the trajectory. It assumes that the terrain is locally flat between the three neighboring image features around the intersection point of the LRF with the ground. Depending on the total number of features tracked and computational power available, the algorithm can elegantly scale from a flat-world assumption (3 features) to a dense 3D map (e.g. 1 feature per pixel). To further constrain estimation position error drift, one solution could be to incorporate pre-mapped landmark measurements but they may not always be available. Since heading error is a key contributor to this position drift, we propose a sun sensor measurement model to observe heading. Section 4 describes the hardware and software environments of our flight testing. This involved range-VIO real-time software execution on a Qualcomm 820 computer board, using commercial off-the-shelf sensors, as envisioned for the science helicopter concept. Flights were performed with a terrestrial drone over analogue Mars terrains. Section 5 discusses results that demonstrate yaw drift elimination with solar-VIO in a Monte Carlo analysis with simulated data, as well as scale drift reduction by a factor of 5 with range-VIO in real time at 30Hz image rate.

2. RELATED WORK

State of the Art in Visual-Inertial Odometry

Monocular Visual-Inertial Odometry (VIO) has become a popular solution for aerial robotics. Even when a GPS signal is available, image measurements can be more accurate when operating in close proximity to surface assets. Moreover, IMUs can provide the metric scale information necessary for controlling flight when the accelerometers are properly excited.

Some of these solutions are based on a loose fusion of the camera with the IMU, where a separate vision-only state estimator provides position and velocity up to scale, and orientation up to local gravity [5]. Loosely-coupled architectures have benefited from the rapid progress of Visual Odometry (VO) in the computer vision community, with algorithms such as PTAM [6], SVO [7] or ORB-SLAM [8]. Even though progress continues to be made with VO, VIO has seen improved results in terms of accuracy and robustness with tightly-coupled approaches, even if they have a higher computational cost [9]. In tight coupling, visual measurements consist of feature tracks or even image intensities, that directly update the inertial estimator. Recent approaches include both filter-based [10] [11] and non-linear optimization-based methods [12] [13]. Some solutions use image feature coordinates for measurements [10] [12] while others optimize on image intensity values [11] [13].

Despite monocular VIO's success in the robotics research community, it still has basic observability limitations for real world applications. First, even under persistent IMU excitation, VIO cannot observe the orientation about the local gravity vector [14], i.e. the heading for UAV applications. Second and most important, the very notion of persistent excitation is lacking in some of the simplest motions that one would expect a UAV to achieve, e.g. to hover, or to fly at uniform velocity. In such cases, VIO cannot observe scale [4], which leads to strong position and velocity drifts.

VIO Drift Mitigation Techniques

Mars Science Helicopter's sensor constraints discussed in the introduction steered our research into the use of range and sun sensing to limit VIO drift.

In our previous work, we designed a range-VIO algorithm for NASA's Mars Helicopter flight project [3]. Unlike most tightly-coupled VIO algorithms, this estimator design was very lightweight with only 21 states. This was achieved through a flat and level-terrain assumption, and a range constraint to constrain the scale drift. Unfortunately, these flat and level terrain assumptions will not hold for Mars Science Helicopter, which is required to visit science targets potentially in rough 3D topographies. On Earth, reference [15] proposed initializing the depth of some VIO features by using measurements from an ultrasonic range sensor. This relaxes the globally-flat terrain constraint to locally-flat, but it also assumes the local terrain slope is perpendicular to the range sensor axis within the ranged area. This will cause issues over 3D topographies, especially given the large beam width of ultrasonic sensors. Many methods use 2D or 3D scanning LRFs in SLAM literature [16]; but to the best of our knowledge, fine terrain modeling for odometry with 1D single-point range finders has not been addressed. In this paper, we propose a measurement model for a laser range finder used alongside a VIO system which assumes the terrain is made of triangular facets with the VIO image features at the edges. Depending on the density of features tracked, this assumption can scale from a flat terrain assumption to virtually no terrain assumption at all in the limit as the number of features tracked increases, i.e. limited only by the available computing power.

Multiple studies have been published which demonstrate terrestrial rover navigation algorithms that augment existing VO algorithms with sun sensors, based on using bundling [17] or Kalman filters [18]. These studies generally found that sun sensor data greatly improved orientation estimates and thus overall position error, with the possible inclusion of inclinometer data in [17]. Sun sensors are also commonly used on spacecraft [19] to help with attitude determination. To the best of our knowledge, sun sensors have not been proposed to complement VIO frameworks.

3. RANGE AND SOLAR VISUAL-INERTIAL ODOMETRY

The state estimation framework illustrated in Figure 1 combines a camera, a LRF, a sun sensor and an IMU to produce low-drift position, velocity and attitude estimates of a Mars Science Helicopter. It is based on an Extended Kalman Filter (EKF), which tightly couples visual, range and solar updates with inertial state propagation.

In this section, we summarize the key design points of our state-of-the-art VIO estimation base, before describing the novel range and solar update models.

Visual-Inertial Odometry

The state vector $\mathbf{x} = [\mathbf{x}_I^T \ \mathbf{x}_V^T]^T$ of our EKF can be divided between the states related to the IMU \mathbf{x}_I , and those related to vision \mathbf{x}_V . The inertial states

$$\mathbf{x}_I = [\mathbf{p}_w^i{}^T \ \mathbf{v}_w^i{}^T \ \mathbf{q}_w^i{}^T \ \mathbf{b}_g^T \ \mathbf{b}_a^T]^T \quad (1)$$

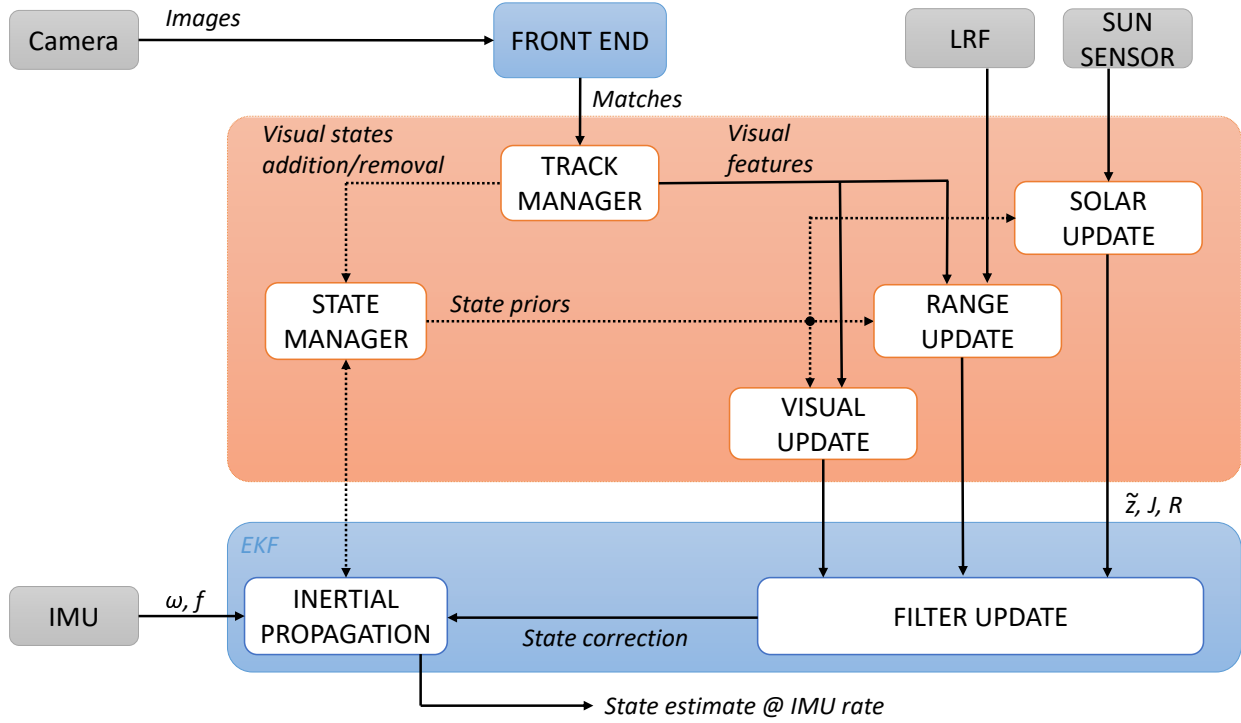


Figure 1: State estimation architecture. The visual, range and solar updates are constructed from the corresponding sensor measurements and state priors. They each deliver a measurement innovation \tilde{z} , a measurement Jacobian matrix J and covariance matrix R that is used by the EKF core to perform a Kalman update. A state management module is responsible for the dynamic construction and destruction of vision states.

include the position, velocity and orientation of the IMU frame $\{i\}$ with respect to the world frame $\{w\}$, the gyroscope biases \mathbf{b}_g and the accelerometer biases \mathbf{b}_a . We use rotation quaternions to model orientations.

The vision states

$$\mathbf{x}_V = \begin{bmatrix} \mathbf{p}_w^{c_1 T} & \dots & \mathbf{p}_w^{c_M T} & \mathbf{q}_w^{c_1 T} & \dots & \mathbf{q}_w^{c_M T} \\ & & & \mathbf{f}_1^T & \dots & \mathbf{f}_N^T \end{bmatrix}^T \quad (2)$$

include the orientations $\{\mathbf{q}_w^{c_i}\}_i$ and positions $\{\mathbf{p}_w^{c_i}\}_i$ of the camera frame at the last M image time instances, along with the 3D coordinates of N visual features $\{\mathbf{f}_j\}_j$. Each feature state $\mathbf{f}_j = [\alpha_j \ \beta_j \ \rho_j]^T$ represents the inverse-depth parametrization of world feature point \mathbf{F}_j with respect to an anchor pose $\{c_{i_j}\}$ selected from the sliding window of pose states.

IMU measurements are used to propagate the state estimate and the error covariance matrix to first order using [20] and [21]. The visual measurement is the pinhole projection of terrain feature \mathbf{F}_j over the normalized image plane $z = 1$ of the camera frame $\{c_i\}$ at time i

$${}^{i,j}\mathbf{z}_{v,m} = \frac{1}{c_i z_j} \begin{bmatrix} c_i x_j \\ c_i y_j \end{bmatrix} + \mathbf{n}_v, \quad (3)$$

where \mathbf{n}_v is a zero-mean white Gaussian feature measurement noise in image space. Equation 3 can be related to the state if we express the cartesian coordinates of feature \mathbf{F}_j in

camera frame $\{c_i\}$ as

$${}^{c_i}\mathbf{p}_j = [{}^{c_i}x_j \ {}^{c_i}y_j \ {}^{c_i}z_j]^T \quad (4)$$

$$= \mathbf{C}(\mathbf{q}_w^{c_i}) \left(\mathbf{p}_w^{c_{i_j}} + \frac{1}{\rho_j} \mathbf{C}(\mathbf{q}_w^{c_{i_j}})^T \begin{bmatrix} \alpha_j \\ \beta_j \\ 1 \end{bmatrix} - \mathbf{p}_w^{c_i} \right). \quad (5)$$

Visual features are detected in the image with the FAST algorithm [22], and tracked with the pyramidal implementation of the Kanade-Lucas-Tomasi algorithm [23] [24].

This subsection only briefly summarized the key design points of our VIO estimator. We refer the reader to [25] and [26] for more complete implementation details.

Range Update

Terrain range measurements are dependent on the poses of the range sensor as well as terrain topography. A range measurement model needs to properly account for uncertainty on both. Since terrain information is included in VIO feature states, we leverage these to construct a new range update model.

Our range update assumes the terrain is locally flat between three SLAM features surrounding the intersection point of the LRF beam with the terrain. For simplification purposes in this paper, we will assume zero translation between the optical center of the camera and the origin of the LRF. This offset can be measured in a calibration step and introduced in the model. Figure 2 illustrates the geometry of the scene. \mathbf{u}_{r_i} is the unit vector oriented along the optical axis of the LRF at time i . \mathbf{I} is the intersection of this axis with the terrain. \mathbf{F}_1 ,

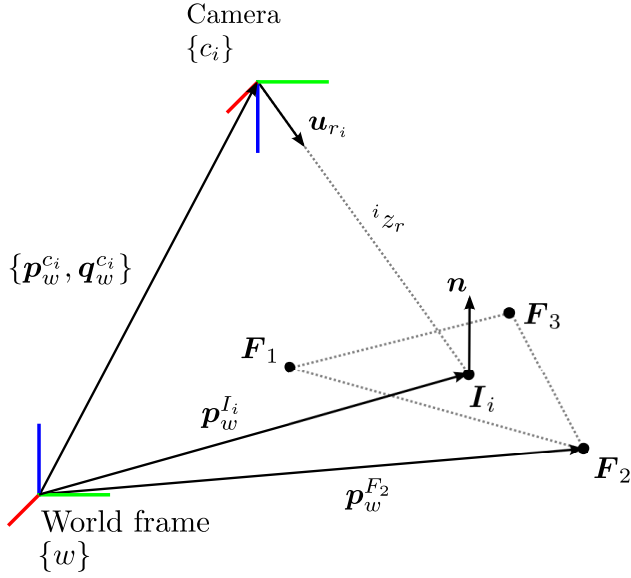


Figure 2: Geometry of the range measurement at time i ${}^i z_r$ and three surrounding visual features F_1 , F_2 and F_3 . u_{r_i} is the unit vector oriented along the optical axis of the LRF, which is assumed to have the same origin as the camera frame $\{c_i\}$. I_i is the intersection of this axis with the terrain.

F_2 and F_3 are SLAM features forming a triangle around I in image space. n is a normal vector to the plane containing F_1 , F_2 , F_3 and I . All the above vectors are assumed to be resolved in the world frame $\{w\}$.

If the dot product $u_{r_i} \cdot n \neq 0$, we can express the range measurement at time i as

$${}^i z_r = {}^i z_r \frac{u_{r_i} \cdot n}{u_{r_i} \cdot n} \quad (6)$$

$$= \frac{(p_w^{I_i} - p_w^{c_i}) \cdot n}{u_{r_i} \cdot n} \quad (7)$$

$$= \frac{(p_w^{F_2} - p_w^{c_i}) \cdot n}{u_{r_i} \cdot n}, \quad (8)$$

where

$$n = (p_w^{F_1} - p_w^{F_2}) \times (p_w^{F_3} - p_w^{F_2}), \quad (9)$$

$\{c_i\}$ is the camera frame at time i , and p_w^* represents the position of an object in the world frame $\{w\}$. Please note that n is not a unit vector in general.

Equation (8) and (9) show the range is a nonlinear function h_r of the state vector x . We assume the LRF measurements suffer from additive zero-mean white Gaussian noise n_r for use in our EKF estimation framework.

$${}^i z_{r,m} = {}^i z_r + n_r \quad (10)$$

$$= h_r(x) + n_r \quad (11)$$

To use this range update in practice, one still needs to select three SLAM features to form the triangle $F_1 F_2 F_3$. In this paper, we perform a Delaunay triangulation in image space over the SLAM features, and select the triangle in which the intersection point of the LRF with the ground falls into.

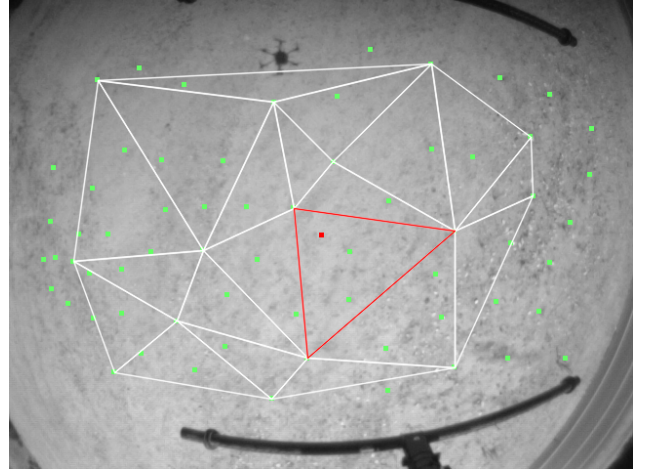


Figure 3: Delaunay triangulation between image features. The features used as filter states form the edges of the triangles. The red dot represents the intersection point of the LRF with the terrain. The surrounding red triangles are used to form the range update. Additional green dots are image features tracked but not used for state estimation.

We selected the Delaunay triangulation since it maximizes the smallest angle of all possible triangulations [27]. This property is interesting to avoid “long and skinny” triangles that do not provide a strong local planar constraint. Figure 3 shows the Delaunay triangulation and triangle selection over a sample flight image from the navigation camera.

Figure 3 highlights the assumption that the terrain surface can be partitioned into triangular facets, with visual features at their edges. It is interesting to note that if the state estimator uses only 3 VIO features in a lightweight fashion, this is equivalent to a globally-flat world assumption. Conversely, when the number of SLAM features increases, the limit of the area of the facets tends to zero and the planar terrain assumption virtually disappears.

Solar Update

Sun sensor measurements are dependent on the orientation of the sun sensor as well as the position of the Sun in the sky. We will ignore issues of cloud cover, since this is not of major concern in the Martian atmosphere. Over short distances and lengths of time, the Sun can also be assumed to be motionless and still provide a global direction. We will do this here, so we do not need to know local sidereal time, the estimated position of the platform on the planet, and Kepler’s laws. We will treat the unit vector pointing from the Sun to the sun sensor, s , as constant. The sun sensor’s measurement, depicted in Figure 4, provides the angles between the sun sensor’s +Z axis and the projection of s onto the XZ and YZ planes.

Given the components x_s , y_s , and z_s of s in the sun sensor frame $\{s\}$, the sun sensor measurement can be written as

$$z_s = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} \arctan\left(\frac{x_s}{z_s}\right) & \arctan\left(\frac{y_s}{z_s}\right) \end{bmatrix}^T \quad (12)$$

where

$$\begin{bmatrix} x_s & y_s & z_s \end{bmatrix}^T = C(q_i^s) C(q_w^i)^w s \quad (13)$$

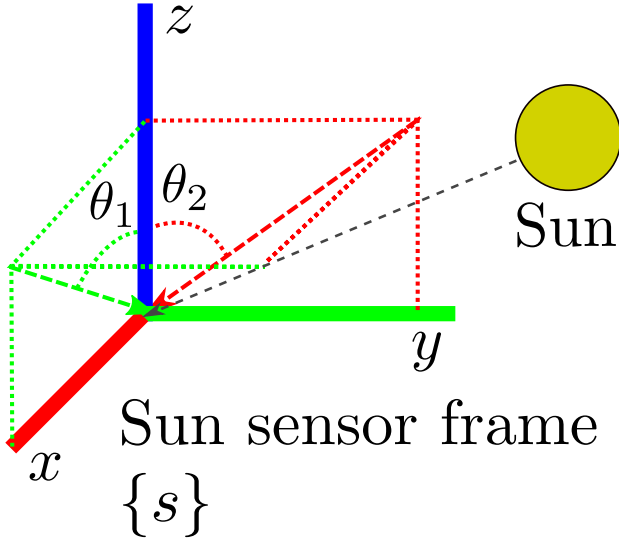


Figure 4: Geometry of the sun sensor measurement $z_s = [\theta_1 \ \theta_2]^T$ and the sun sensor frame $\{s\}$. The sun vector s is drawn as a gray dotted line connecting the origin of $\{s\}$ to the sun. The green and red dotted lines demonstrate the orthogonal projection of s onto the XZ and YZ planes. The polar angles of these projections define θ_1 and θ_2 .

Equation (12) shows the sun sensor measurement is a nonlinear function of the state vector, which can be integrated into our EKF estimation framework.

$$z_s = h_s(x) + n_s \quad (14)$$

4. TESTING

We deployed our algorithm on a UAV that was equipped with an avionics payload to execute the range-visual-inertial state estimator presented in Section 3 in real-time while simultaneously logging all relevant sensor data for off-line analysis.

UAV platform

The UAV consisted of a TurboAce Infinity6 hexacopter shown in Figure 5 that was commanded in an autonomous fly-to-waypoint mode using GPS-based closed-loop controls. The on-board autopilot was Pixhawk Cube with the Ardupilot firmware. A typical trajectory included take-off to a specific altitude, fly a waypoint pattern, return to the take-off position, and land.

Sensors

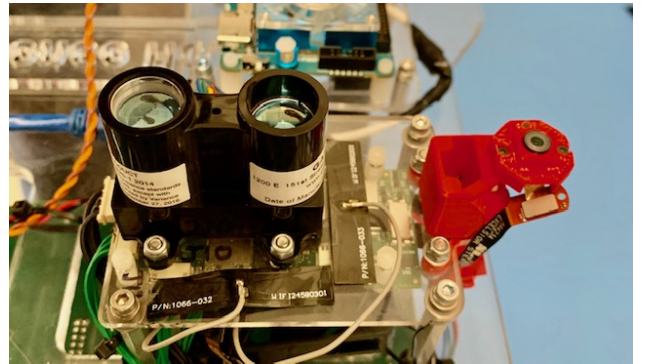
Grayscale navigation images were provided by an Omnivision OV7251 camera, with a resolution of 640×480 , depth of 8 bits and auto-exposure. Range data came from a Garmin Lidar Lite V3 single-point static range finder, which has an accuracy of 2.5 cm and a range of 40 m. Inertial data were delivered by STIM300 tactical grade IMU. No specific vibration isolation setup was used for the IMU. Flight accelerometer data show vibration amplitude levels up to 10 m.s^{-2} , which is one order of magnitude higher than much lower grade IMUs, such as Pixhawk's MPU9250 in our tests. We are currently investigating these issues. All the sensors can be seen in Figure 6.



Figure 5: TurboAce Infinity6 Hexacopter with avionics payload used for flight testing.



(a) Top panel with STIM300 IMU and TMS570 microcontroller



(b) LRF and camera

Figure 6: Avionics payload used during flight testing. The upper image shows the top panel of the avionics board, with the STIM300 IMU (in orange) and the TMS570 microcontroller (in red). The lower image shows the LRF sensor (two black cylinders) and the navigation camera (on the right).

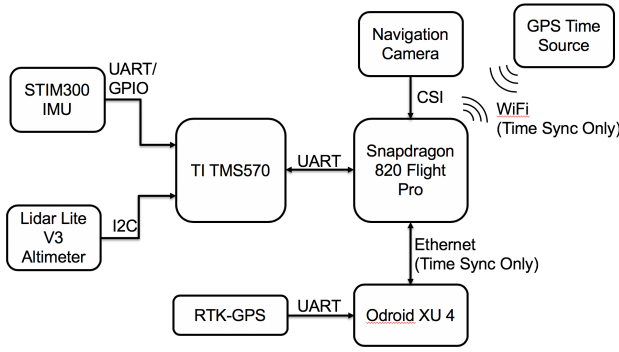


Figure 7: Avionics system diagram

Real-time environment

The avionics payload is designed as a stand-alone module representative of future MSH avionics. The main processor is a Snapdragon 820, a quad-core 64-bit ARM CPU, with 2 high power cores clocked at 2.1 GHz, and two high efficiency cores clocked at 1.6 GHz. The Snapdragon 820 is responsible for executing the navigation algorithm and logging avionics sensor data. A Texas Instruments TMS570 microcontroller accurately timestamps IMU and range data and passes the sensor to the Snapdragon 820 over a serial bus. The Snapdragon 820 directly reads camera images over a Camera Serial Interface. All avionics sensor data is logged to disk in addition to being published over a ROS network. An Odroid XU4 is included for logging RTK-GPS data. The RTK-GPS is directly logged to Odroid memory during the test flights. The Snapdragon 820 is time synchronized to a GPS time source and the Odroid is time synchronized to the Snapdragon 820. A system diagram of the avionics payload is provided in Figure 7.

The Snapdragon 820 and TMS570 run custom software for sensor data handling written using the F-Prime framework [28]. F-Prime is a modular software framework tailored to supporting small scale spaceflight systems. The navigation software C++ implementation is currently executed in a ROS environment for research purposes on one high power core of the Snapdragon 820.

Ground Truth

Position was provided by a RTK-GPS system composed of a Trimble BD930-UHF receiver with a BX982 base station. This setup provides centimeter accuracy in outdoor environments. It also served as time server for all of our time synchronization. GPS coordinates were averaged before take-off to set the local origin, and converted to East (E), North (N), Up (U) coordinates from that reference point.

Position estimates are readily time-synced with the RTK-GPS ground truth. The origin of the state estimator's world frame $\{w\}$, introduced in Section 3 was also set to the take-off location. The axes of $\{w\}$ are defined as the IMU axes at the time of state estimator initialization, which typically are not aligned with GPS' ENU. To compute position error, position estimates were interpolated to the times of GPS "ground truth" measurements, and a 3D rotation centered on the common origin was computed through a least-square fit [29].

Attitude and velocity ground truth are work in progress and not available in our current setup. Since such attitude ground

truth is needed to characterize the yaw drift reduction using sun sensor measurements, we instead focused on analysing the performance of solar-VIO completely by computer simulation, as will be presented in the next section.

5. RESULTS

Solar-VIO on simulated data

In the absence of attitude ground truth on real datasets, we performed a 1000-run Monte Carlo analysis of our solar-VIO algorithm on simulated data to characterize yaw drift reduction performance. The reference trajectory for these runs is 3 minutes long, with the craft moving at 4 m/s in a circular pattern of 10-m radius, 5 m above the simulated ground, with the body rotating in the horizontal plane at the same angular rate that it moves. Run-to-run randomness came from sensor noise, which was set to be representative of a SolarMEMS ISS-D60 sun sensor with 0.06-deg precision that we have characterized on real data, our visual front end performance and a Pixhawk MPU9250 IMU. Inertial data was simulated at 200 Hz, solar and image at 20Hz.

Figures 9 and 8 compare the attitude and position errors, respectively, for VIO with and without solar update.

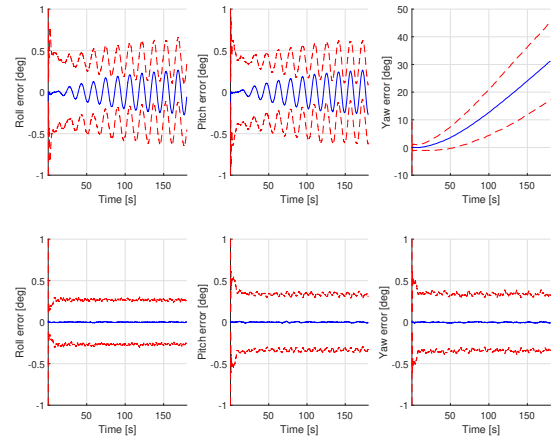


Figure 8: Plots of the attitude estimation errors in world frame, with (below) and without (above) the sun sensor. Mean error is plotted in blue, with dashed red lines indicating 3-sigma bounds calculated from the data.

Figure 8 clearly shows an impressive drift error reduction in attitude. Although one would expect only yaw drift from VIO, attitude errors also grow over the x and y axes in world frame because of the position drift along the circle path, observed in Figure 9, that we assume due to the lack of sufficient VIO excitation. The non-zero mean error in the VIO case were related to linearization errors during IMU integration, due to the circular motion, the high velocity and IMU frequency. Solar VIO is able to reduce the standard deviation significantly on all axes, and eliminate the error biases.

Range-VIO on Real Flight Data in Real Time

The hardware platform described in Section 4 was flown in the Arroyo Seco next to JPL in Pasadena, CA.

During these flights, our state estimator was running in real

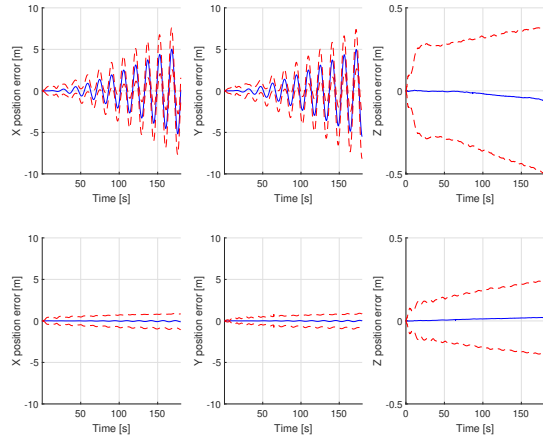


Figure 9: Plots of the position estimation errors in world frame, with (below) and without (above) the sun sensor. Mean error is plotted in blue, with dashed red lines indicating 3-sigma bounds calculated from the data.

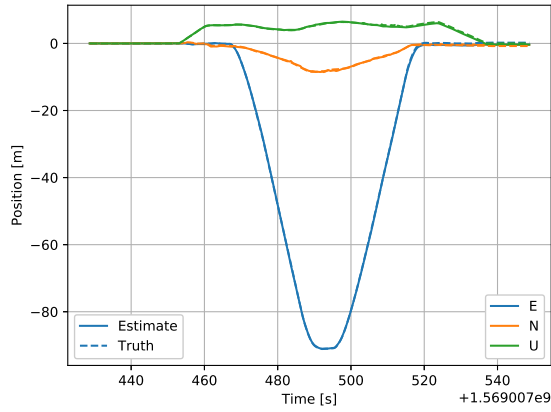


Figure 10: Real-time range-visual-inertial flight position estimate, compared to ground truth. Position components are expressed in the East (E), North (N), Up (U) frame at origin, which is the origin of the IMU frame at the start of the flight.

time on one of the high-performance cores of the Snapdragon 820 board. Image and range data was processed at 30Hz, and inertial data at 250Hz. No sun sensor data was used during this tests. The visual front end was set to track at least 50 features, up to 15 of which were used for VIO. No specific IMU excitation was provided before or during flight. Apart from control disturbances and dynamic transients, the trajectory was programmed to achieve a straight uniform velocity profile between waypoints.

Figure 10 shows a comparison between the estimated and true position on a representative 90-meter flight commanded at 6-m altitude and 5-m/s horizontal speed. The position error is shown in Figure 11. The maximum position error per axis is 0.6 m, and below 1 m in norm. This is less than 0.5% of the distance traveled, which is on par with best VIO performance under proper excitation [30].

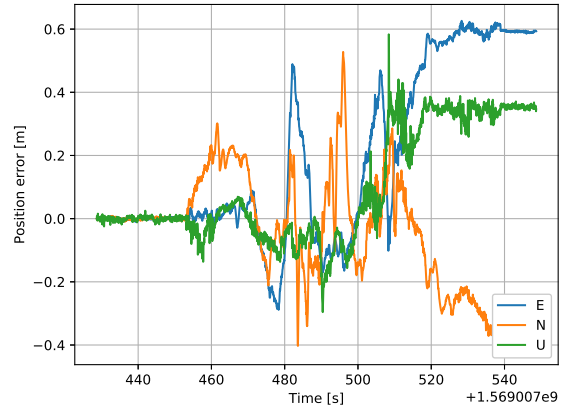


Figure 11: Real-time range-visual-inertial position error components. Position components are expressed in the East (E), North (N), Up (U) frame at origin, which is the origin of the IMU frame at the start of the flight.

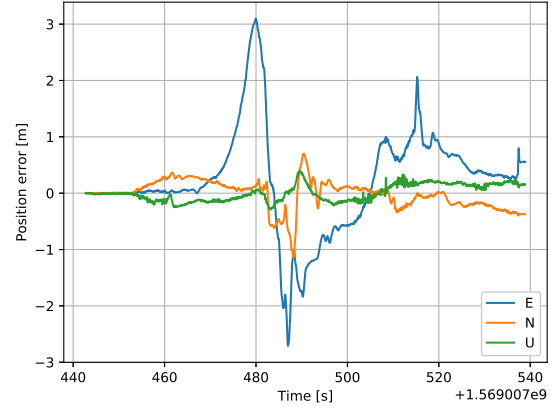


Figure 12: Visual-inertial position error components. Position estimates were obtained offline by replaying the sensor data recorded during the real-time flight of Figure 11, to allow for direct comparison of the benefit of using range measurements in addition to visual-inertial only. Position components are expressed in the East (E), North (N), Up (U) frame at origin, which is the origin of the IMU frame at the start of the flight.

To provide an explicit comparison, we re-ran the estimator offline on the visual and inertial data recorded during the same flight, but this time without using range measurements. Figure 12 shows the VIO position errors. It rises up to 3 meters in the East axis, which is the direction of travel. Compared to the range data case, the position error is 5 times bigger on that axis. This result is consistent with the observability properties of VIO reviewed in Section 2, which claims that scale is not observable under uniform velocity, and hence the drift will be larger along the direction of travel.

It is interesting to correlate the errors with the terrain topography to verify our assumptions. Figure 13 shows the LRF measurements during the same flight. To first order, these can be considered equivalent to height above terrain. The ascent

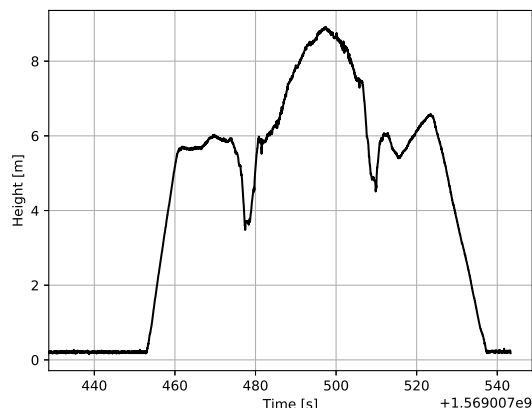


Figure 13: Terrain range measurements from downward-pointing laser range finder. In the absence of attitude ground truth, these are assumed to be equivalent at first order to height (above terrain) measurements along the local vertical.

phase finishes at $t = 460$ s, and the descent starts at $t = 525$ s. In between, the flight is controlled to be horizontal. Height above terrain varies rapidly between +3.5 m and +9 m, with two mounds visible at $t = 475$ s and at $t = 510$ s. This terrain was selected to test our topography assumption. It seems to be well modeled since the growing position estimation error during the horizontal part of the flight interval is always under 0.6m in Figure 11. Local error peaks are under investigation and may be related local triangular facet violation, or require further tuning.

6. SUMMARY

We presented a modular state estimator design based on visual, range, solar and inertial measurements for a future Mars Science Helicopter concept. Such a mission requires low-drift localization to reach science targets far apart with a limited set of sensors above the surface of Mars. We introduced a novel range update model to constrain VIO scale drift using a single-point range finder. It is based on a triangular-facet terrain assumption, that can be scaled from flat-terrain to virtually no-topography-assumption at all based on the number of VIO features processed, or equivalently the computational power available. Range-VIO was demonstrated in real time onboard a Qualcomm Snapdragon 820 processor in flight. This processor is the successor of the Snapdragon 801, which will launch for the first airborne powered flight on another planet in July 2020 with Mars Helicopter. We also incorporated a sun sensor measurement model to constrain VIO yaw drift. Sun sensor benefits were demonstrated in a Monte Carlo analysis on simulated data.

Future work include characterization of range VIO errors on steeper and vertical terrain slopes, integration of attitude ground truth into our real-time datasets to help characterize solar VIO yaw drift reduction on real datasets, VIO using range and solar observations together, and using the estimator for supporting closed-loop controls.

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 (80NM0018D0004).

© 2019 California Institute of Technology. Government sponsorship acknowledged.

REFERENCES

- [1] H. Grip, J. Lam, D. Bayard, D. Conway, G. Singh, R. Brockers, J. Delaune, L. Matthies, C. Malpica, T. Brown, A. Jain, A. San Martin, and G. Merewether, "Flight Control System for NASA's Mars Helicopter," in *AIAA Scitech Forum*, 2019.
- [2] J. Balaram, T. Canham, C. Duncan, M. Golombek, H. Grip, W. Johnson, J. Maki, A. Quon, R. Stern, and D. Zhu, "Mars Helicopter Technology Demonstrator," in *AIAA Scitech Forum*, 2018.
- [3] D. Bayard, D. Conway, R. Brockers, J. Delaune, L. Matthies, H. Grip, G. Merewether, T. Brown, and A. San Martin, "Vision-Based Navigation for the NASA Mars Helicopter," in *AIAA Scitech Forum*, 2019.
- [4] K. Wu and S. Roumeliotis, "Unobservable directions of vins under special motions," University of Minnesota, Tech. Rep., 2016.
- [5] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli, and R. Siegwart, "Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 957–964.
- [6] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, 2007.
- [7] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2014.
- [8] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.
- [9] J. A. Delmerico and D. Scaramuzza, "A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots," *International Conference on Robotics and Automation (ICRA)*, pp. 2502–2509, 2018.
- [10] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2007, pp. 3565–3572.
- [11] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct ekf-based approach," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.
- [12] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.
- [13] L. von Stumberg, V. C. Usenko, and D. Cremers,

- “Direct sparse visual-inertial odometry using dynamic marginalization,” *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2510–2517, 2018.
- [14] D. G. Kottas, K. J. Wu, and S. I. Roumeliotis, “Detecting and dealing with hovering maneuvers in vision-aided inertial navigation systems,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 3172–3179.
- [15] S. Urzua, R. Munguía, and A. Grau, “Vision-based slam system for mavs in gps-denied environments,” *International Journal of Micro Air Vehicles*, vol. 9, no. 4, pp. 283–296, 2017.
- [16] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” *Trans. Rob.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.
- [17] A. Lambert, P. Furgale, T. D. Barfoot, and J. Enright, “Visual odometry aided by a sun sensor and inclinometer,” in *2011 Aerospace Conference*. IEEE, 2011, pp. 1–14.
- [18] A. Trebi-Ollennu, T. Huntsberger, Y. Cheng, E. T. Baumgartner, B. Kennedy, and P. Schenker, “Design and analysis of a sun sensor for planetary rover absolute heading detection,” *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 939–947, 2001.
- [19] J. Wertz and L. W.J., *Space Mission Analysis and Design*, 3rd ed. Springer, 2007.
- [20] S. Weiss and R. Siegwart, “Real-time metric state estimation for modular vision-inertial systems,” in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2012.
- [21] N. Trawny and S. Roumeliotis, “Indirect kalman filter for 3d attitude estimation,” University of Minnesota, Tech. Rep., 2005.
- [22] E. Rosten, R. Porter, and T. Drummond, “Faster and better: A machine learning approach to corner detection,” *Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.
- [23] J.-Y. Bouguet, “Pyramidal implementation of the lucas kanade feature tracker,” *Intel Corporation, Microprocessor Research Labs*, 2000.
- [24] J. Shi and C. Tomasi, “Good features to track,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 1994, pp. 594–600.
- [25] J. Delaune, R. Hewitt, L. Lytle, C. Sorice, R. Thakker, and L. Matthies, “Thermal-inertial odometry for autonomous flight throughout the night,” in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2019.
- [26] J. Delaune and D. Bayard, “xvio: Visual-inertial odometry implementation,” Jet Propulsion Laboratory (available upon request), Tech. Rep., 2019.
- [27] B. Gärtner and M. Hoffmann, “Computational geometry lecture notes hs 2013,” ETH Zürich, Tech. Rep., 2013.
- [28] “NASA F-Prime Flight Software Framework,” <https://github.com/nasa/fprime>, accessed: 2019-10-15.
- [29] K. Arun, T. Huang, and S. Blostein, “Least-squares fitting of two 3-d point sets,” *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 9, no. 5, pp. 698–700, 1987.
- [30] M. Li and A. I. Mourikis, “High-precision, consistent ekf-based visual-inertial odometry,” *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, 2013.

BIOGRAPHY



Jeff Delaune received his Ph.D. in Robotics from Institut Supérieur de l’Aéronautique et de l’Espace (ISAE, France) in 2013, after a M.S. in Astronautics and Space Engineering from Cranfield University (United Kingdom), and a B.S./M.S. in Engineering from École Centrale de Nantes (France). He is currently a robotics technologist at the Jet Propulsion Laboratory. He develops autonomous navigation systems for planetary exploration, with a focus on flying vehicles. Jeff was part of the navigation team for NASA’s Mars Helicopter flight project. His interests include state estimation, sensor fusion, perception and computer vision.



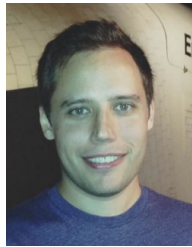
Roland Brockers is a Research Technologist at the Jet Propulsion Laboratory. He received his Ph.D. in Electrical Engineering from the University of Paderborn, Germany in 2005, and has been conducting research in autonomous navigation of unmanned robotic systems for more than 19 years with a focus on aerial vehicles since 2010 where he worked on autonomous landing and ingress, visual pose estimation, and autonomous obstacle avoidance for micro air vehicles. He was part of the JPL Mars Helicopter GNC team and wrote the image-processing software for the Mars Helicopter flight project on Mars 2020. He is the PI of the Advanced Navigation for future Mars Rotorcraft R&D effort. His current research interests include 3D perception systems for small mobile robotic platforms, and autonomous robotic systems with applications in earth science and planetary applications.



David S. Bayard is currently a JPL Technical Fellow, with 35 years of experience in the aerospace industry. During the period 2017-2018, Dr. Bayard led the JPL Mars Helicopter Navigation Team responsible for developing the first autonomous vision-based navigation system designed to fly on board a planetary drone. His work includes over 170 papers in refereed journals and conferences, 63 NASA Tech Brief Awards, and 4 U.S. patents. Dr. Bayard received numerous NASA Awards and Medals during his career, and received the American Automatic Control Council’s (AACC) Control Engineering Practice Award. He is an Associate Fellow of AIAA.



Harel Dor is currently a senior at Caltech pursuing B.S. degrees in Applied Physics and Computer Science. They worked with the NASA's Mars Science Helicopter team over the summer of 2019 through Caltech's Summer Undergraduate Research Fellowship. Their interests include space exploration, embedded systems, and automation.



Robert Hewitt received his Ph.D. degree in Electrical and Computer Engineering in 2018 from Queen's University (Kingston, Canada), after a M.A.Sc. in Aerospace Engineering from Carleton University (Ottawa, Canada) and a B.A.Sc in Engineering Physics from the University of Saskatchewan (Saskatoon, Canada). He is currently a post-doctoral research scientist at the Jet Propulsion Laboratory. He develops autonomous navigation systems for aerial and ground vehicles used in prospective planetary missions. His interests include state estimation, perception, and instrumentation.



robotics.

Jacek Sawoniewicz is a Robotics Systems Engineer in the Robotic Mobility Group at the NASA Jet Propulsion Laboratory, California Institute of Technology. He received his M.S. degree in Electronics Engineering and Robotics from Warsaw University of Technology, Faculty of Electronics and Information Technology in 1999. His main interests are hard real-time software and space



Gerik Kubiak received his B.S. degree in Computer Engineering in 2016 from the California Polytechnic State University - San Luis Obispo. He currently works as a robotics system engineer for Mars 2020 Robotic Arm. Previously he worked on software and operations for the Mars Helicopter flight project. His interests include embedded systems and robotics.



Theodore Tzanetos received his B.S. degree in Electrical Engineering and Computer Science in 2012 and a M.Eng. in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology in 2013. He is currently a test conductor for the Mars Helicopter flight project on Mars 2020 and PI for the Mars Science Helicopter System R&D effort. His interests include embedded systems, aerial vehicles, and autonomy.



Larry Matthies is a Senior Research Scientist at the Jet Propulsion Laboratory. He has conducted research there on computer vision-based autonomous navigation of robotic vehicles for 30 years. He defined initial versions of several vision algorithms that have been used on Mars or are their way to Mars for the 2020 mission, including stereo vision and visual odometry for rover navigation, feature tracking for velocity estimation during terminal descent in the Descent Image Motion Estimation System for the MER mission, and terrain relative navigation for Mars precision landing, which is now built into the 2020 Mars landing system. He has also been a Principal Investigator in many autonomous navigation research programs funded by DARPA and the U.S. Army. He obtained his PhD in computer science from Carnegie Mellon University in 1989 and is a Fellow of the IEEE.



J. (Bob) Balaram is Principal Member of Technical Staff at the NASA Jet Propulsion Laboratory where he is with the Mobility & Robotic Systems Section. He received his Ph. D. in Computer and Systems Engineering from Rensselaer Polytechnic Institute. At JPL he has been active in the area of telerobotics technology development for Mars Rovers, planetary balloon aerobot systems, and multi-mission, high-fidelity Spacecraft simulators for Entry, Descent and Landing and Surface Mobility. Bob was the chief Engineer for the Mars Helicopter project at the Jet Propulsion Laboratory.