Design and Analysis of Map Relative Localization for Access to Hazardous Landing Sites on Mars

Andrew E. Johnson[1], Seth Aaron[2], Yang Cheng[3], James Montgomery[2] Nikolas Trawny[4], Brent Tweddle[4], Geoffrey Vaughan[2] and Jason Zheng[5]
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109

Human and robotic planetary lander missions require accurate surface relative position knowledge to land near science targets or next to pre-deployed assets. In the absence of GPS, accurate position estimates can be obtained by automatically matching sensor data collected during descent to an on-board map. The Lander Vision System (LVS) that is being developed for Mars landing applications generates landmark matches in descent imagery and combines these with inertial data to estimate vehicle position, velocity and attitude. This paper describes recent LVS design work focused on making the map relative localization algorithms robust to challenging environmental conditions like bland terrain, appearance differences between the map and image and initial input state errors. Improved results are shown using data from a recent LVS field test campaign. This paper also fills a gap in analysis to date by assessing the performance of the LVS with data sets containing significant vertical motion including a complete data set from the Mars Science Laboratory mission, a Mars landing simulation, and field test data taken over multiple altitudes above the same scene. Accurate and robust performance is achieved for all data sets indicating that vertical motion does not play a significant role in position estimation performance.

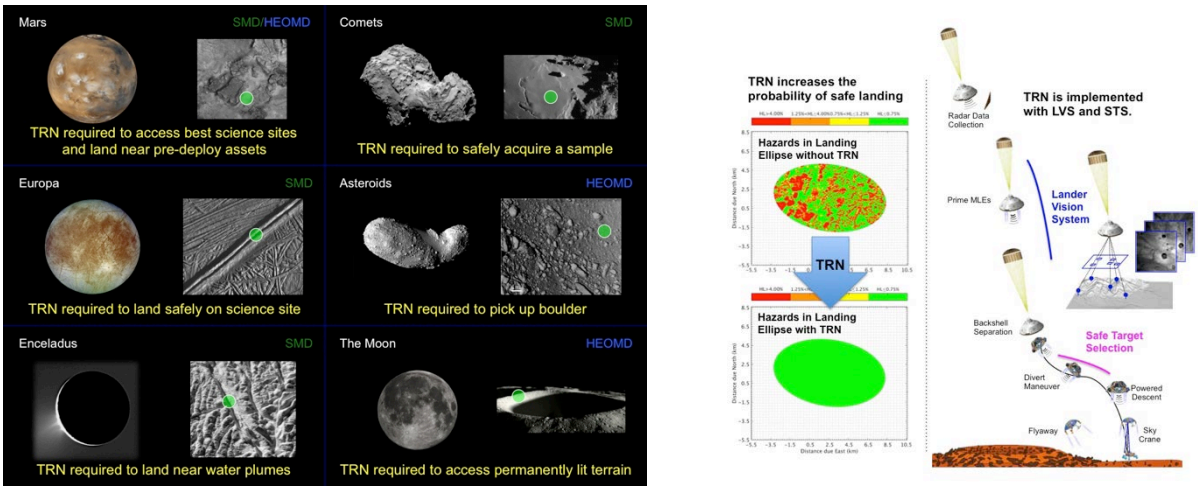
I. Introduction

Terrain Relative Navigation (TRN) is an on-board spacecraft GN&C function that determines the position of a spacecraft relative to a map of a planetary surface. Given adequate divert capability this measurement can be used to guide the vehicle to the center of the landing ellipse. If the divert capability is limited then the position estimate can still be used to avoid hazards identified in the map prior to landing. The latter approach can be added to the Mars Science Laboratory (MSL) Entry Descent and Landing system [7] to avoid 100m scale hazards in the landing ellipse which will enable selection of landing sites that have scientifically interesting terrain relief and were not selectable in the original implementation of MSL EDL. Mars scientists see great value in adding this capability to the proposed Mars 2020 lander mission [3][5]. As shown in Figure 1a, TRN can also be used for robotic and human missions to the surface of comets, asteroids, Europa, Enceladus and the Moon.

TRN technology development at JPL has been ongoing for more than a decade. Recent work has focused on development of a flight-like prototype called the Lander Vision System (LVS) which was flown in a real-time helicopter field test conducted in February and March 2014 [2]. During this test, less than 40m position knowledge errors were demonstrated over a wide variety of terrain, illumination conditions, and attitude dynamics. This successful proof of concept led to a study to develop the initial design of TRN for the Mars 2020 mission. It turns out that TRN can be implemented with very little impact to the MSL EDL architecture that M2020 is inheriting. As shown in Figure 1b, the LVS can be added to the EDL system and operated only during the parachute descent phase. The position estimate is

[1] Principal Member of Tech Staff, Guidance & Control Section, MS 198-235, 4800 Oak Grove Dr., AIAA Member.
[2] Member of Tech Staff, Guidance and Control Section, MS 198-235, 4800 Oak Grove Dr.
[3] Principal Member of Tech Staff, Mobility and Robotic System Section, MS 198-235, 4800 Oak Grove Dr.
[4] Member of Tech Staff, Guidance and Control Section, MS 198-235, 4800 Oak Grove Dr., AIAA Member.
[5] Member of Tech Staff, Flight Electronics and Software Systems Section, MS 198-326, 4800 Oak Grove Dr.

provided to Safe Target Selection (STS) which would be a small augmentation to the MSL EDL Guidance Navigation and Control software. STS uses the position estimate and an onboard map of hazards to determine a target point that avoids hazards and the path of the back-shell. This new target point is handed of to the heritage MSL guidance and control system to fly the vehicle to the safe landing site.



(a) (b)

Figure 1. (a) TRN has broad applicability across human and robotic missions within NASA. (b) The Mars 2020 implementation of TRN combines the Lander Vision System with a Safe Target Selection algorithm to avoid known hazards in the landing ellipse.

The LVS is composed of a camera, Inertial Measurement Unit (IMU) and a dedicated compute element loaded with real-time software and FPGA firmware to perform state estimation at a 1Hz image update rate (Figure 2). The LVS hardware has a straightforward path to flight qualification: major processor parts have flight qualified equivalents, 5Hz mega-pixel cameras have flown in space (e.g., the MSL MARDI descent imager) and IMUs are common. The LVS runs Map Relative Localization (MRL) algorithms which fuse landmark matches between pixels in a descent image and 3D locations in a map with inertial angular rates and accelerations provided by the IMU. The MRL algorithms are broken into landmark matching which makes heavy use of the FPGA for image processing and state estimation which runs on the processor. Landmark matching is broken into a coarse phase that takes out the initial large position error (up to 3km for Mars 2020) and a fine phase that improves the position accuracy to a less than 40m. The landmark matching and state estimation over many images must finish in 10 seconds to have minimal impact on the Mars 2020 EDL timeline.

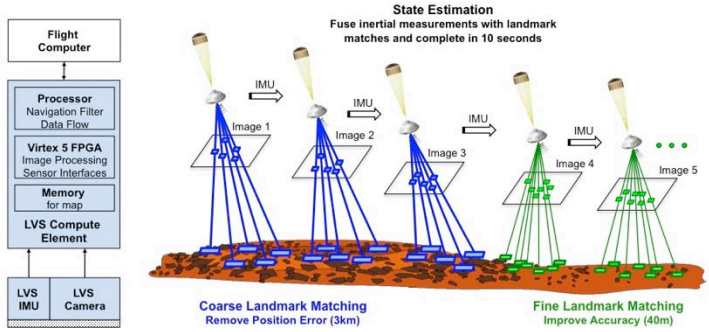


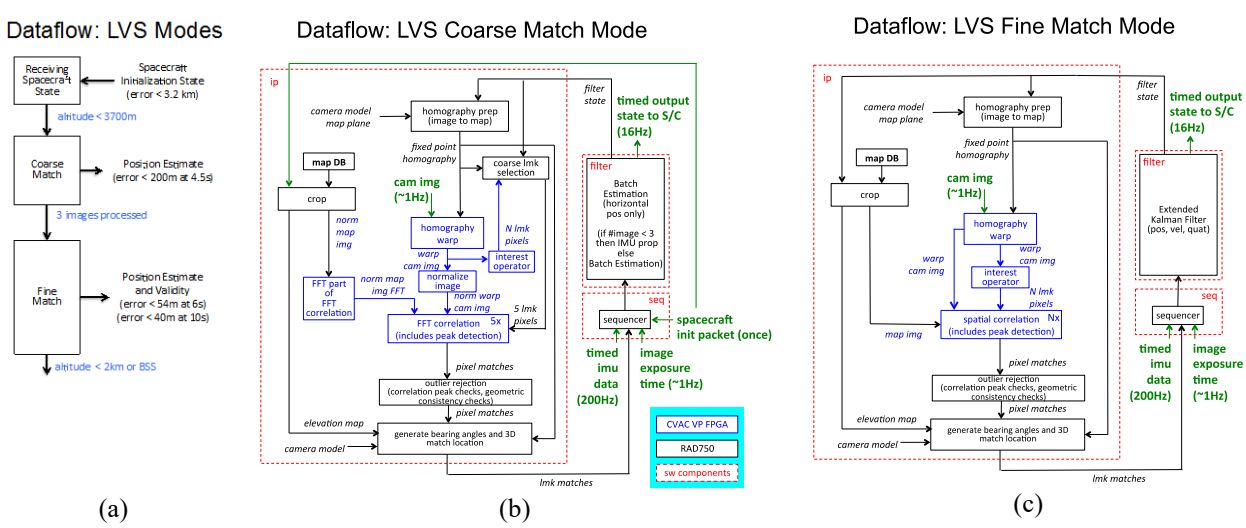
Figure 2. LVS hardware components and processing phases with expected requirements for Mars 2020.

II. Map Relative Localization Algorithm Improvements

The MRL algorithms are organized in a data flow architecture (Figure 3); each step in the algorithm can only execute when the all of the required inputs are available. This type of architecture precludes the need for an executive to control data flow which simplifies communications and makes optimal use of processing time. The messaging does not use time, so speeding up or slowing down input message rate produces a corresponding speed up or slowdown in the whole system – assuming sufficient processing capability. The only time-dependent part is the sending of the final estimate to the spacecraft, but this is independent of the estimation process. Hence, the MRL code is time-independent.

The MRL code has three modes (Figure 3a). It starts in a Receiving Spacecraft State Mode which waits to start until messages arrive with the spacecraft position (with large error), attitude, velocity and altitude. This mode monitors the messages and once the altitude is less than 3700m for the Mars 2020 application, the MRL code switches the Coarse Match Mode (Figure 3b). In this mode, image warping and frequency based image correlation are run on the FPGA to create landmark matches between the image and map. Once three images are processed, with five landmarks each, the landmarks are passed to a Batch Estimation processing that computes a position correction. At this point the MRL code enters the Fine Match Mode (Figure 3c). For each image in this mode, landmark matches are generated, using image warping and spatial correlation on the FPGA, then passed to an Iterative Extended Kalman Filter which updates the position, velocity, attitude and IMU biases. Fine Match Mode ends when the altitude goes below 2km or the back-shell separates (BSS) which signifies the end of the parachute descent phase and the start of powered descent. At this point STS takes the final valid position estimate and uses it to identify the safe landing site to target. Both coarse and fine landmark matching make extensive use of five image processing modules implemented on a Xilinx Virtex 5 FPGA. The modules are shown in blue in Figure 3 while typical run-times that enable on 1Hz image update rate are given in Table 1.

The MRL processing has been described in detail in previous publications [2][4] however some important changes have been recently implemented. The changes in order in which they occur in the processing flow are described in the detail in the section below.



(a) (b) (c)

Figure 3. Map Relative Localization has a coarse matching mode followed by a fine matching mode and utilizes a standard processor and an FPGA. The MRL code is broken into three software components (ip, seq, filter) that cut across modes. (a) MRL modes, (b) coarse match mode and (c) fine match mode.

A. Coarse Landmark Selection

Recent improvements to FFT Correlation and Homography Warp have decreased their run-time to the point where it is now possible to match five landmarks in each coarse match image in the amount of time it previously took to match three coarse landmarks. Separately, analysis of some field test runs showed

that landmark matching had large errors (> 200m) when landmarks were selected with a small amount of high contrast terrain on the edge of the landmark. This problem could be mitigated by moving the landmark over to be centered on the high contrast terrain. The increase in landmarks and the need to incorporate image data motivated a change to the coarse landmark selection method as described below.

The image is warped to the map plane as usual using Homography Warp. The Interest Operator FPGA core is applied to the warped image to identify high contrast pixels in the warped image. The coarse feature selector takes these inputs and selects five landmarks by first selecting a 3x3 grid of landmarks that are within the image border and the map (black squares in Figure 4a). It then shifts each landmark to be centered on the Interest Operator pixel with highest contrast contained within the corresponding black square (gray and green squares in Figure 4a). It then iteratively picks the five landmarks. It looks at the highest contrast landmark. If it does not overlap with the opposition effect (gray circle in Figure 4a) or any other already selected landmarks then it is selected. This process repeats with the remaining landmarks. If five landmarks cannot be selected then the landmark overlap constraint is eliminated. As shown by the five green landmarks in Figure 4a this results in a wide spread of landmarks across the image that are also placed on high contrast terrain.

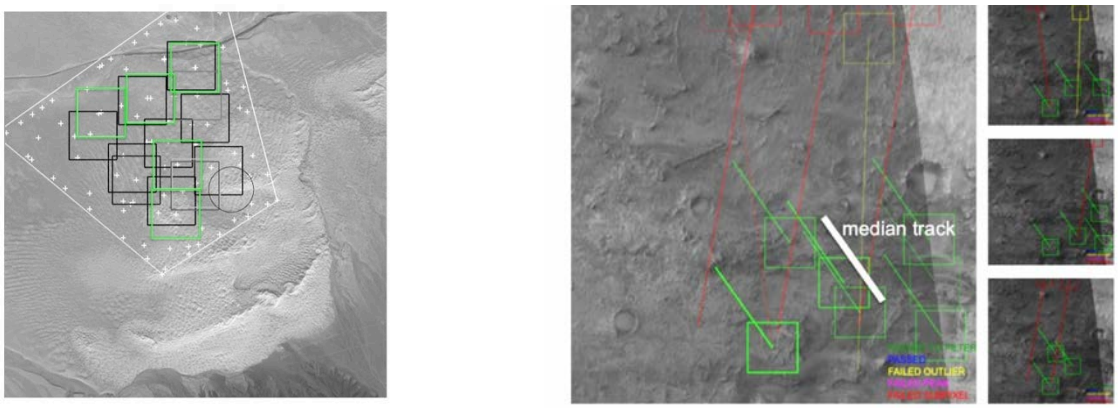
Table 1. FPGA image processing modules that enable 1 Hz MRL update rates.

Name	Description	Coarse Mode	Fine Mode	Run Time on COTS Virtex 5 FPGA
Homography Warp	Warp planar image to a view from a new position and attitude.	X	X	0.156s for 1024x1024 image
Image Normalization	Subtract mean and divide by variance in 21x21 pixel local image window	X		0.218s for 1024x1024 image
FFT Correlation	2D correlation of two images in the frequency domain with peak detection	X		0.113s for 1024x1024 image
Interest Operator	Select pixels with strong brightness gradients in orthogonal directions (modified Harris)	X	X	0.062s to find 256 landmarks
Spatial Correlation	2D spatial correlation of an image template with an image window with peak detection		X	0.165s for 127 landmarks and 100 pixel search window

B. Coarse Landmark Outlier Rejection

Image Normalization and FFT Correlation occurs have not changed from the prior implementation. However outlier rejection has changed to mitigate issues seen in some field test results. The main issue was that the prior implementation of outlier rejection worked on each image separately; with a small number of landmarks in each image there were cases where there was only one inlier. The previous outlier rejection could not distinguish this inlier from the outliers so it was thrown out as well. Often this happened in all three coarse images which resulted in no coarse landmark matches and ultimately no position estimate.

It turns out that for coarse match, because they all have the same large position error, all of the correct landmark matches from all of the images should have consistent residuals between the predicted location of the landmark in the map and the matched location in the map (these are the line segments emanating from the squares in Figure 4b). If a landmark match does not follow these correct tracks then it is an outlier. Using this observation it is possible to make a more robust outlier rejection scheme as follows. First, matches that overlap the map border or have correlation peaks that are too wide or too short are eliminated (the red landmarks in Figure 4b). The medians of delta row and delta column of the remaining residuals is computed (median track). All matches with residuals that are within a threshold in row and column of this median track are inliers (the green landmarks in Figure 4b) and the rest are outliers (the yellow landmark in Figure 4b). Since the batch estimation does not process the landmark matches until all 3 images have been processed this change in approach does not have a significant effect on the processing architecture.



(a) (b)

Figure 4. (a) Coarse landmark selection combines geometric constraints and the descent image appearance to select landmarks. (b) Coarse landmark outlier rejection looks at the correlation peak shape and height and the landmark match residuals to differentiate correct from incorrect matches.

Reducing the state will make the Batch Estimator less sensitive to outliers, so a study was conducted to determine which states could be eliminated. The initial attitude estimate and the IMU gyros have very low noise , so IMU propagation during coarse match can be used to provide attitude to fine match. Similarly, the altitude and velocity estimates errors are low (< 100m in altitude, < 2 m/s in velocity) so IMU propagation also works for providing those states to fine match. IMU biases are expected to be very small, so they do not need to be estimated at all. The only remaining state that needs to be estimated in coarse match is horizontal position.

As shown in Figure 5, the Batch Estimator was modified to estimate a single horizontal position shift $\Delta p_{h,0}$ of the first image; the propagated attitude, altitude and delta position between each image are fixed by IMU propagation. The position shift is applied to the propagated trajectory and this trajectory is passed to fine match. Batch Estimation takes 96ms on the LEON3 processor in the LVS prototype.

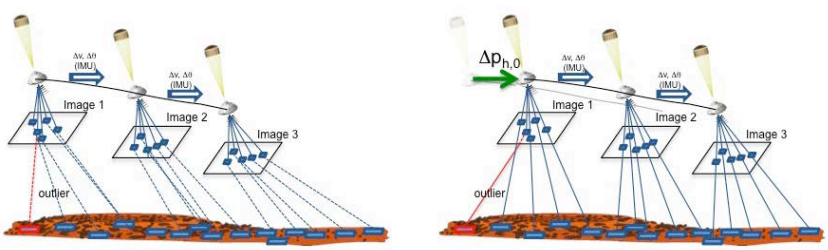


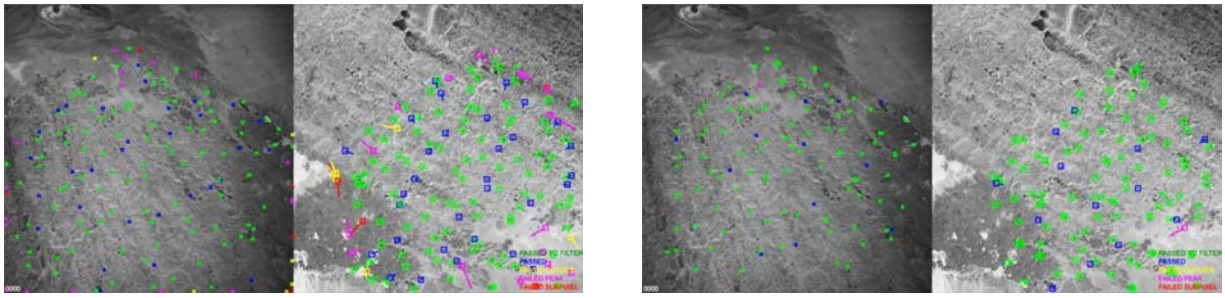
Figure 5. A single shift in position is estimated in the coarse match phase using landmark matches from all images (up to 5 from each of 3 images) and the inertial data between images.

C. Fine Match Interest Operator Selection

The prior Interest Operator worked on the descent image after it was warped into the map frame. The 1024x1024 map array with the warped image data in it was split into a 16x16 grid of 64x64 pixel windows. In each window the Interest Operator selected the pixel with the highest contrast. This simple implementation works well except for windows near the warped image border. These locations have more radial distortion so are less likely to match correctly. There is also a known issue where, because it operates on 5x5 pixels, the Interest Operator can select pixels that are within 2 pixels of the image data border. When these selected pixels are passed to Spatial Correlation, parts of the 15x15 pixel correlation template could go outside the image data and end up not matching correctly.

A quick fix to both of these problems was to have a function on the processor compute the position of the 64x64 pixel image windows in which to apply the interest operator. The current function uses the

regular 16x16 grid and just eliminates the windows that intersect the image border. Figure 6a shows a fine match with the original code. Notice that many matches near the bottom and sides of the image are thrown out by outlier rejection and therefore were not good for matching. In Figure 6b landmarks are not selected near the border and therefore there are much fewer incorrect matches. Preventing incorrect matches from entering the system from the beginning places less of a burden on outlier rejection and is good design practice. Given this new capability to select interest operator search windows on the fly it could be used to solve other off nominal conditions like known hot pixels and known foreground objects in the field if view or even dynamic conditions like selecting features closer to nadir.



(a) (b)

Figure 6. (a) The prior interest operator code could select landmarks on the edge of the image border which resulted incorrect matches (red, pink and yellow matches). (b) A new interest operator selection approach avoids the border of the image which results in fewer outliers and less processing.

D. Fine Match Extended Kalman Filter

The Kalman filter used in the fine match mode is not new, but describing it is important for understanding the interactions between the LVS and the spacecraft in the proposed Mars 2020 implementation. The Coarse Match Mode provides a horizontal position correction with less than 200m error that updates propagated state from the space craft initialization message (Figure 3a). At the beginning of Fine Match Mode an image is taken and passed to image processing (Figure 7). Meanwhile the state is propagated forward using IMU data and the current state estimate. When the image processing is completed (approximately 600ms later) up to 100 landmark matches are passed into an Iterative Extended Kalman Filter (IEKF) which performances a state update of position, velocity, attitude and IMU biases. The IEKF takes 4 iterations and completes in 162ms on the LEON3 processor. During the IEKF processing the LVS state is held constant; after the update, the processor quickly propagates the IMU forward from the time the image was taken which induces a moderate (< 500ms) latency on the state estimates passed to the spacecraft. After six seconds of processing the LVS has taken 3 coarse and 2 fine images and the position estimate is typically less then 54m in error and confidence in the answer is high enough to send to the spacecraft (i.e., valid). At ten seconds 3 coarse and 5 fine image have been processed and the full 40m position error has been achieved.

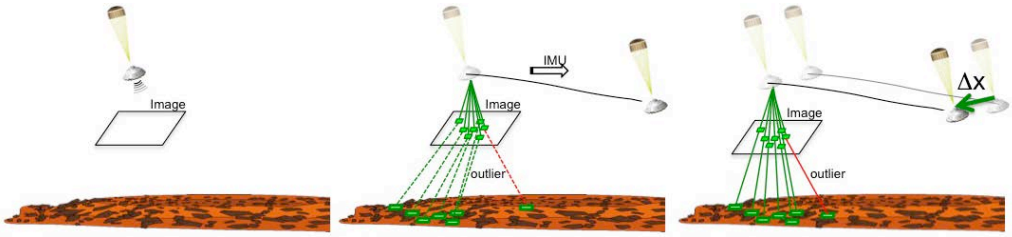


Figure 7. In the fine match mode the entire lander state is updated using many landmark matches from a single image. To deal with image processing latency, the state is propagated forward using IMU data from the time the image was taken. Once the update is available all of the states since the image exposure are corrected.

III. Field Test Performance Analysis with Improvements

Over the last year the LVS field test runs were processed by the LVS compute element to assess the various performance improvements. Figure 8 shows the position errors at the three key stages of processing for the 125 runs collected during the field test that were in the Mars 2020 operational envelope[6]. At the end of coarse match the horizontal error is less than the 200m requirement while the vertical error is still as large as the initial altitude input error (+/- 100m). After 5 images (i.e, 6 seconds) the horizontal error is less than the 54m requirement and the vertical error has reduced to less than +/- 5m. Finally after 8 images (i.e., 10 seconds) the horizontal error is less than the 40m requirement and the vertical error has not changed dramatically. The colors on the plots correspond to the total number of landmarks matches through all the images: Green more than 600, blue between 600 and 500, cyan between 400 and 500, yellow between 300 and 400, orange between 200 and 300 and red less than 200.

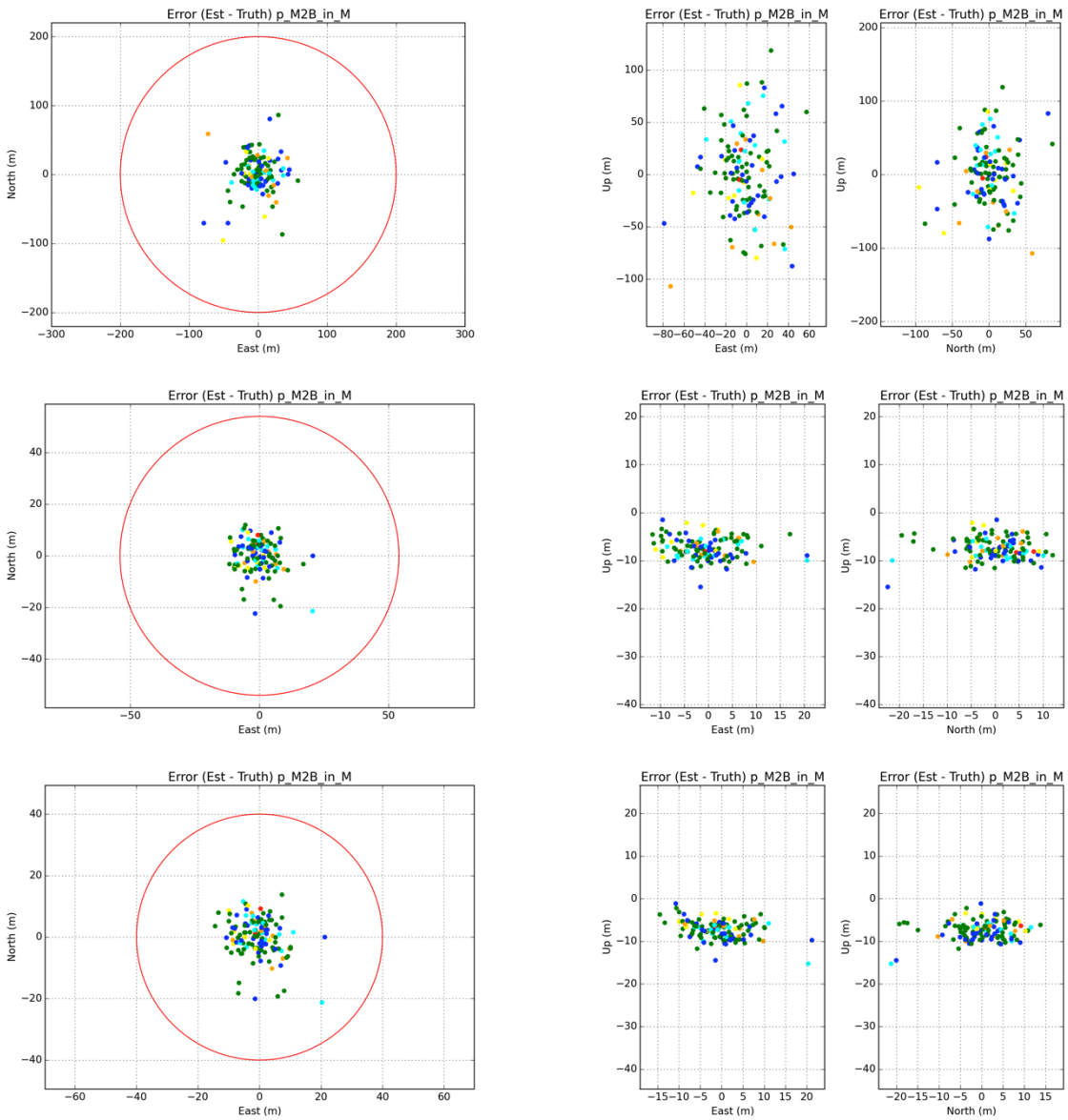
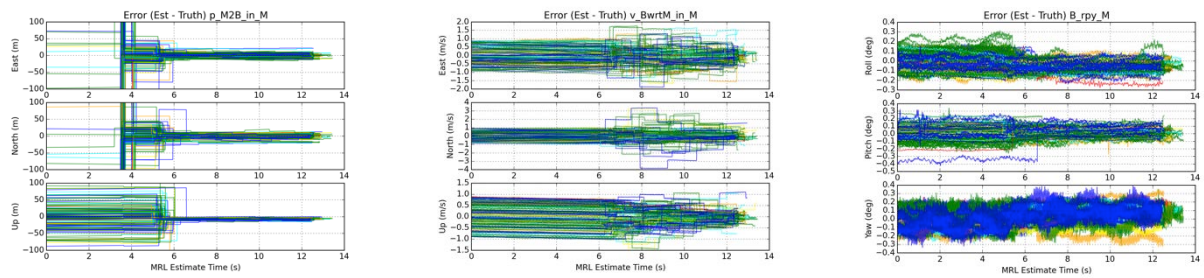


Figure 8. Field test horizontal and vertical position errors after coarse match (top row), after coarse match and 2 fine images (middle row) and after coarse match and 5 fine images (bottom row).

[6] Three of the 128 LVS runs reported in [2], were later determined to not be in the Mars 2020 operational envelope.

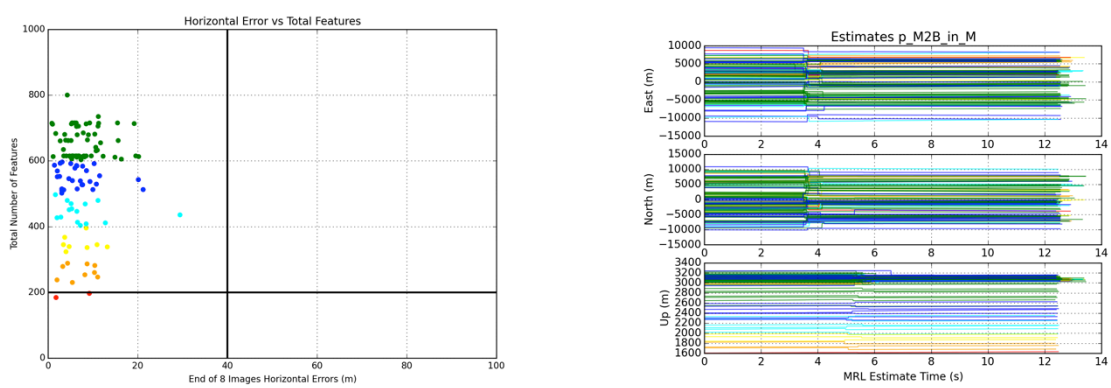
Figure 9 shows the position, velocity and attitude errors of the state estimate with respect to the time the MRL estimate is valid; due to image processing latency and IMU catch up this is not real-time. The position errors show the decrease over time that is expected. The velocity errors stay at the constant error of the initial spacecraft estimate then jump around between +/-2m/s (<0.1% of altitude) in all axes after fine match starts and velocity is updated in the state. The velocity errors have a blocky shape because image updates drive the errors. Attitude errors are less than +/- 0.3 degrees. The attitude errors should be constant during the coarse match phase since attitude is not being updated, however that is not the case. This discrepancy is thought to be either a small timing or misalignment error. It would be difficult to obtain a small position error and also have small velocity and attitude errors. Therefore, because the velocity and attitude errors are small, they could be compared to spacecraft velocity and attitude estimates to confirm the position estimate is valid. If the velocity and attitude do not agree with the spacecraft estimates then the position is likely incorrect as well.

Figure 10a shows the total number of landmarks matched with respect to horizontal error. For these runs there is not a strong dependence between horizontal error and number of landmarks. Instead as shown in Figure 10b the number of landmarks is tightly correlated with the altitude (i.e., Up) of the LVS run. This is expected because the number of fine landmarks selected is proportional to the area of the image footprint. As image footprint shrinks with altitude so does the number of fine landmarks.



(a) position error (b) velocity error (c) attitude error

Figure 9. Position, velocity and attitude errors from the 125 field test runs.



(a) (b)

Figure 10. (a) Comparison of final horizontal error to the total number of landmarks tracked in through all images. (b) Position estimate shows that altitude (Up) is highly correlated with total numbers of landmarks.

With each algorithmic change, the 125 field test runs were processed by the LVS prototype compute element to assess the performance improvement. Particular attention was paid to the three runs that had a low number of total landmarks during the field test. The impact of the algorithm changes on these runs is shown in Table 2. Initially all runs matched landmarks in coarse match but did not find any fine matches. The first simple fix, not described above, was to make the thresholds on Spatial Correlation peak height and width more restrictive. This change was significant enough that it enabled two of the runs to obtain

fine matches. Changing to the horizontal position only batch estimation enabled the final run to obtain fine matches. Subsequent changes had moderate impact until the update to the Fine Match Interest Operator which boosted the number of landmarks for the two bland terrain runs significantly.

The statistical position errors from all 125 runs are also shown in Table 2. The horizontal errors have standard deviations around 6m which is consistent with the 6m/pixel maps used in the fine match phase. The horizontal position errors have an appreciable bias of around 3m which (around half a map pixel) until the landmark data structure was updated to fix errors in the map cropping process. After this fix the bias was essentially zero as can be seen in the plots in Figure 8. Throughout all the fixes the vertical position has a bias of around 7m and a standard deviation around 2.5m. The source of this elevation bias is unknown but could possibly an error in the calibrated focal length of the camera; 7m at 3km altitude could result from a 0.2% error in focal length.

Table 2. The effect of recent algorithmic improvements on three of the most challenging field test runs.

Fix			11/18/14	3/28/15	4/6/15	8/24/15	10/5/15	11/18/15
2014 Fixes			X	X	X	X	X	X
Spatial Correlation Threshold Tuning				X	X	X	X	X
Position Only Batch Init Estimator					X	X	X	X
5 Coarse Features						X	X	X
New Landmark Definition							X	X
Border Avoiding Interest Operator								X
#	**Run**	**Issue**	**Total Number of Matches**					
1	Amboy AM01.2.24a	big change in appearance between map and image	6	141	129	201	182	184
2	Cadiz CZ01.1.07a	bland next to high contrast terrain	9	203	212	183	180	247
3	Cadiz CZ01.2.11a	bland next to high contrast terrain	6	9	145	192	191	282
Statistic			**Error (m)**					
Mean East Error			-1.1	3.8	3.9	3.3	0.1	-0.6
Mean North Error			-3.0	-2.3	-2.3	-2.4	0.1	0.2
Mean Up Error			-7.6	-7.5	-7.1	-6.9	-7.1	-7.4
Standard Deviation East Error			6.0	6.2	6.2	5.9	6.2	6.0
Standard Deviation North Error			6.0	5.6	5.9	6.3	6.4	6.3
Standard Deviation Up Error			2.5	2.5	2.3	2.4	2.3	2.3

IV. Analysis of the Effect of Vertical Motion on Map Relative Localization

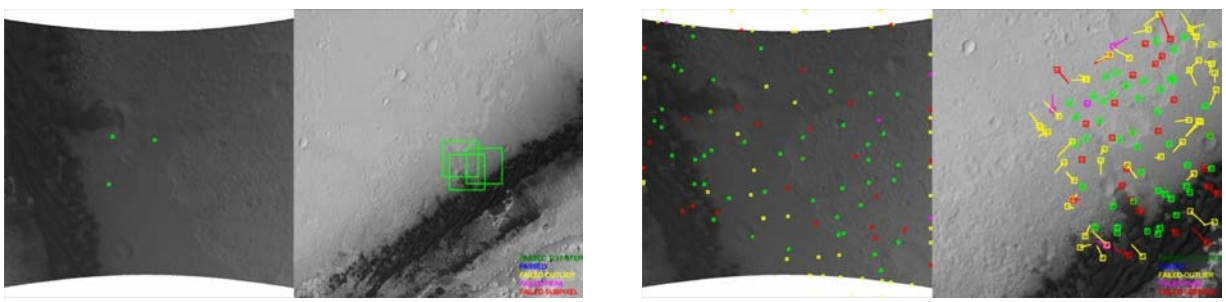
Typical Mars descents have vertical velocity between 60 and 115 m/s while the helicopter used in the LVS field test could only descend at a few meters per second. Because it was not possible to test the effect of typical vertical motion on MRL performance in the field test three different studies were conducted to understand this effect: MSL Data Analysis, DSENDS Simulation Analysis and Interpolated Field Test Descent Analysis.

A. MSL Data Analysis

MSL collected a complete data set for MRL analysis including MARDI [6] camera descent images, Descent IMU (DIMU) data and a reconstructed trajectory for ground truth estimation. Although the data was collected, the system was not designed to synchronize the timing of the MARDI and DIMU data and to further complicate matters the ~4Hz MARDI images were only time-tagged internally to the nearest second. Before the analysis could proceed, significant effort went into aligning the two data sets in time by generating trajectories of each independently and then correlating them to obtain a single time offset.

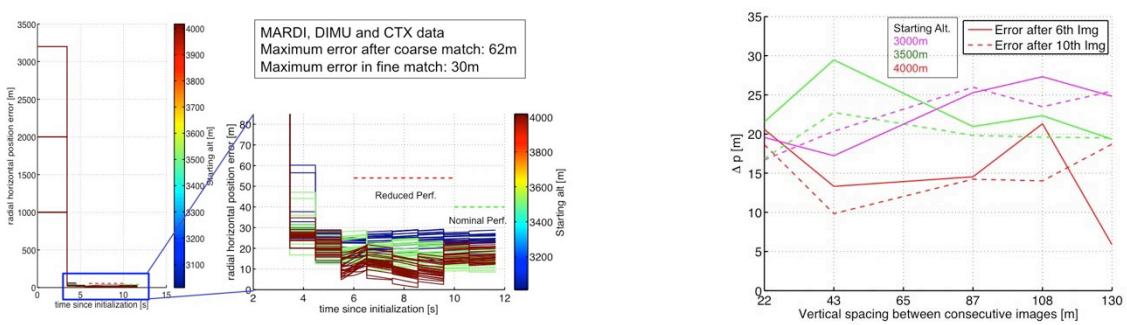
Once the data set was prepared, it was "replayed" into the LVS compute element to estimate positions using a map generated from MRO CTX imagery. Example coarse and fine match results between MARDI on the left and the CTX map on the right are shown in Figure 11. A variety of studies were performed to show the sensitivity to altitude, time between images and initial position error. Figure 12a shows the horizontal position estimation performance for image sequences starting at different altitudes and also with different initial position errors. The plot shows that MRL easily meets its requirements on position error (red = 54 m from 6s to 10s or processing, green = 40m from 10s on) at different starting altitudes and that it is not sensitive to initial position error.

The MARDI frame rate was around 4Hz. LVS processing occurs at about 1Hz so the analysis above used every 4th image with a change in altitude around 85m between images. In a subsequent analysis the position estimation performance was investigated for images taken with different steps between images (every image, every 2nd, 4th, 5th and 6th image) as a way to investigate sensitivity to vertical motion. As shown in Figure 12b the position estimation performance was not sensitive to vertical motion.



(a) (b)

Figure 11. Example MRL landmark match between MARDI and MRO CTX imagery. The right in each image is a rectified MARDI image with the location of the landmarks while the right is the MRO CTX map. (a) Example coarse landmark matches and (b) example fine landmark matches. The green landmark matches are passed to the estimator while the red, purple and yellow landmarks are as outliers that are not passed.



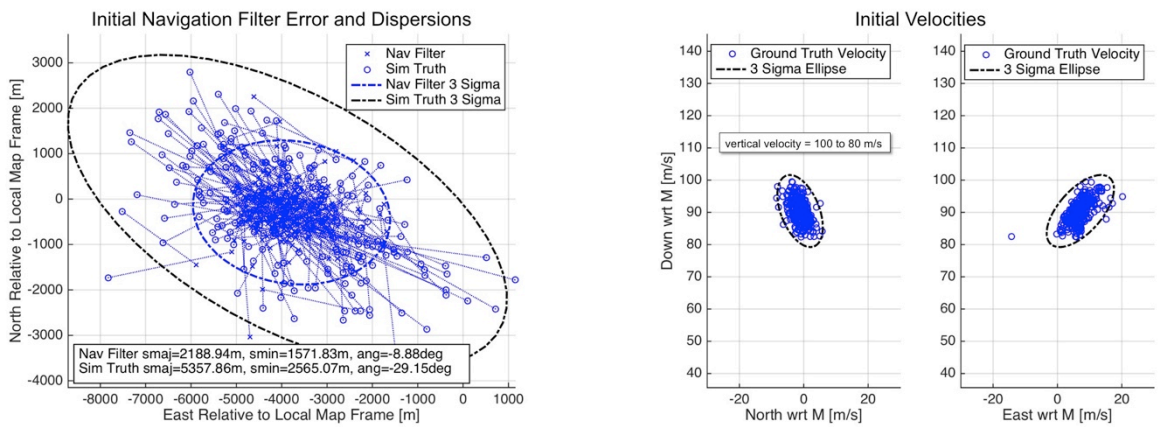
(a) (b)

Figure 12. (a) MSL data starting altitude sensitivity study. When in the operational altitude regime expected for Mars 2020, LVS meets the position estimation accuracy requirements (red and green lines) with margin. (b) A comparison of different performance with different time between images for the MSL data shows that vertical motion does not have a significant influence on horizontal position errors.

B. DSENDS Simulation Analysis

The second set of analysis used completely synthetic data to explore vertical motion sensitivity. The DSENDS simulation [1] used by MSL and updated for M2020 was modified to output DIMU data and camera images taken during parachute descent to the North East Syrtis landing site. The camera images were generated using an existing tool that renders a terrain map textured with an image. The renderer adds some atmospheric effects and camera noise representative of the proposed camera for LVS on Mars 2020. These data sets were input to the LVS compute element using a TCP/IP socket interface that provided the data at the appropriate time for processing.

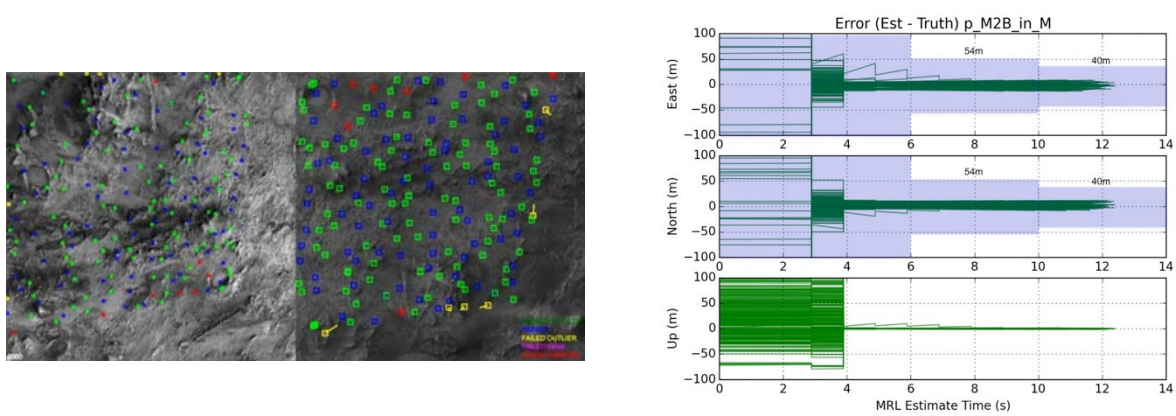
A 250 run Monte Carlo was generated to test MRL with a variety of trajectories, positions in the ellipse and initial position knowledge errors. Figure 13a shows the dispersions of the trajectories across the ellipse as circles and the position error as the line segment to the blue X. Figure 13b shows the dispersions of velocities between 80 and 100 m/s vertical



(a) (b)

Figure 13. DSENDS simulation of 250 different trajectories across the a Northeast Syrtis landing site. (a) Horizontal dispersions and position errors and (b) velocity dispersions.

A typical fine match between rendered image and CTX map are shown in Figure 14a. The position errors after LVS processing are shown in Figure 14b. All of the runs meet the horizontal position estimation requirements (shown in blue shading) and all of the runs had a large number of matches landmarks although it should be noted that the camera simulation is only of moderate fidelity and does not have all noises and non-ideal imaging effects. Regardless these results also show that the LVS is not effected by the vertical motion in the data set.



(a) (b)

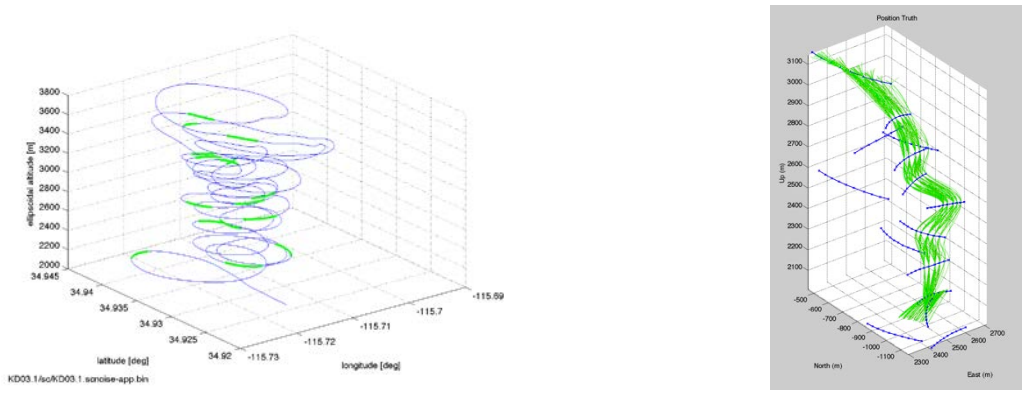
Figure 14. DSENDS simulation with LVS results. (a) A typical fine match result. (b) the position errors for all 250 Monte Carlo runs.

C. Field Test Descent Analysis

The MSL analysis was a very realistic assessment of MRL with a real Mars data set, but it was only one descent. The DSENDS simulation generated multiple descents but with simulated and potentially overly optimistic inputs. Ideally the analysis would use multiple descents with real data. To partially fill in this gap the field test descent analysis used real images from the LVS field test taken at a range of altitudes combined with simulated IMU data generated from a trajectory that would go through all the image position and attitudes.

Two of the flights over Kelso Sand Dunes had ascending spirals with LVS runs every 2 minutes or so during the ascent. The LVS runs covered altitudes from 1.6km up to 3.2km with a nominal spacing of 100m in altitude. A single image was selected from almost every run and then stitched together with a spline trajectory that preserved the position and attitude at each image time. The trajectory was then differentiated to generate angular rates and accelerations measured by an IMU. Noises were then added to simulate real IMU data. Because the images had large changes in position and attitude the accelerations and angular rates were much larger than would actually happen during descent. Figure 15 shows one of the ascending spirals and a set of trajectories generated from the LVS runs.

For each spiral, 127 runs were synthesized and "replayed" through the LVS compute element. The scatter plot of horizontal position errors is shown in Figure 16. The figure also shows a scatter plot of the horizontal position errors for the LVS runs during the horizontal helicopter flights that were used to generate the descending sequences. The two scatter plots agree quite well indicating that vertical motion does not have a large effect on horizontal position estimation.



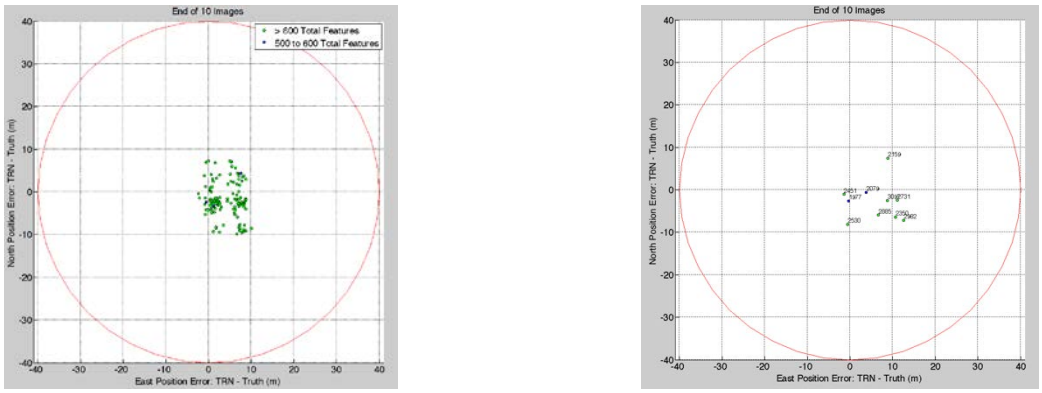
(a) (b)

Figure 15. (a) One of the spiral ascents over Kelso Sand Dunes with LVS operation shown in green. (b) Example sequence of trajectories generated that stitch together images from multiple LVS runs.

V. Conclusion

Multiple improvements to the Lander Vision System Map Relative Localization algorithms were implemented and tested with data collected from the LVS field test. These algorithm improvements eliminated the previously reported bias in the field test horizontal position errors while also turning previous invalid runs into valid runs.

All three vertical motion data sets show LVS meets the 40m horizontal position estimate performance requirement. The MSL and Field test analyzes indicate that vertical motion does not have a significant effect on position estimation performance. This is expected because (a) accelerometer errors do not depend on velocity direction, (b) image quality (blur) is not effected by vertical velocity and (c) landmarks matching is a function of altitude and attitude knowledge and not velocity.



(a) (b)

Figure 16. Scatter plot of final horizontal position errors for 127 runs constructed from one ascending spiral (left) compared to the scatter plot of position errors from the original horizontal flight helicopter runs. The agreement shows that vertical motion does not have a significant effect on position estimation performance.

Acknowledgments

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. This work was funded by the Mars Program in the NASA Science Mission Directorate and the JPL Research and Technology Development Program.

References

[1] P. D. Burkhart, J. Casoliva, B. Balaram, "Mars Science Laboratory entry descent and landing simulation using DSENDS," Proc. *23rd AAS/AIAA Space Flight Mechanics Meeting,* AAS-13-148, 2013

[2] A. Johnson, Y. Cheng, J. Montgomery, N. Trawny, B. Tweddle, and J. Zheng, "Real-Time Terrain Relative Navigation Test Results from a Relevant Environment for Mars Landing," *Proc. AIAA Guidance, Navigation, and Control Conference*, January 2015.

[3] A. Johnson and M. Golombek, Lander Vision System for Safe and Precise Entry Descent and Landing, *Concepts and Approaches for Mars Exploration,* June 12–14, 2012, Houston, Texas. LPI Contribution No. 1679.

[4] A. Johnson, C. Bergh, Y. Cheng, et al., "Design and Ground Test Results for the Lander Vision System," *36th Annual AAS Guidance and Control Conference*, AAS 13-042, 2013

[5] M. Golombek., et al., 2nd Landing Site Workshop for the 2020 Mars Rover mission *by the Mars Exploration Program Analysis Group (MEPAG)*, URL http://marsnext.jpl.nasa.gov/workshops/wkshp_2015_08.cfm. [cited 18 Nov 2015].

[6] M. Malin et al., "The Mars Science Laboratory (MSL) Mars Descent Imager (MARDI) flight instrument," *40th Lunar and Planetary Science Conference,* Extended Abstract 1199, Lunar and Planetary Institute, 2009.

[7] A. Steltzner, A. M. San Martin, T. Rivellini, "Mars Science Laboratory Entry, Descent and Landing System Development Challenges and Preliminary Flight Performance," *23rd AAS Space Flight Mechanics Meeting*, AAS-13-123 2013.