Super Resolution Image Enhancement for a Flash Lidar: Back Projection Method

Alexander Bulyshev[a], Glenn Hines[b], Michael Vanek[b], Farzin Amzajerdian[b], Robert Reisse[b] and Diego Pierrottet[c]

[a]AMA Inc., Hampton, VA 23681
[b]NASA Langley Research Center, Hampton, VA 23681
[c]Coherent Applications, Inc., Hampton, VA 23681

ABSTRACT

In this paper a new image processing technique for flash LIDAR data is presented as a potential tool to enable safe and precise spacecraft landings in future robotic or crewed lunar and planetary missions. Flash LIDARs can generate, in real-time, range data that can be interpreted as a 3-dimensional (3-D) image and transformed into a corresponding digital elevation map (DEM). The NASA Autonomous Landing and Hazard Avoidance (ALHAT) project is capitalizing on this new technology by developing, testing and analyzing flash LIDARs to detect hazardous terrain features such as craters, rocks, and slopes during the descent phase of spacecraft landings. Using a flash LIDAR for this application looks very promising, however through theoretical and simulation analysis the ALHAT team has determined that a single frame, or mosaic, of flash LIDAR data may not be sufficient to build a landing site DEM with acceptable spatial resolution, precision, size, or for a mosaic, in time, to meet current system requirements. One way to overcome this potential limitation is by enhancing the flash LIDAR output images. We propose a new super-resolution algorithm applicable to flash LIDAR range data that will create a DEM with sufficient accuracy, precision and size to meet current ALHAT requirements. The performance of our super-resolution algorithm is analyzed by processing data generated during a series of simulation runs by a high fidelity model of a flash LIDAR imaging a high resolution synthetic lunar elevation map. The flash LIDAR model is attached to a simulated spacecraft by a gimbal that points the LIDAR to a target landing site. For each simulation run, a sequence of flash LIDAR frames is recorded and processed as the spacecraft descends toward the landing site. Each run has a different trajectory profile with varying LIDAR look angles of the terrain. We process the output LIDAR frames using our SR algorithm and the results show that the achieved level of accuracy and precision of the SR generated landing site DEM is more than adequate for detecting hazardous terrain features and identifying safe areas.

Keywords: super-resolution, image enhancement, flash lidar, back projection

1. INTRODUCTION

The term super-resolution (SR) applied to image reconstruction, as opposed to optical diffraction limited SR techniques, generally refers to the group of signal processing algorithms that improve the spatial resolution of an image to better than the size of the pixel of the original image. This technique is implemented by taking advantage of sub-pixel shifts between multiple, low-resolution images of the same scene. Historically, SR algorithms were developed for two-dimensional (2-D) image processing.[1–5] However, a new imaging technology called flash LIDARs (lidars) can generate, in real-time, 2-D intensity data, and more pertinent, range data that can be interpreted as a three-dimensional (3-D) image. Flash lidars enable a new set of applications. For example the NASA Autonomous Landing and Hazard Avoidance (ALHAT) project is developing, testing and analyzing flash lidars to detect hazardous terrain features such as craters, rocks, and slopes during the descent phase of spacecraft landings.[6] Through theoretical and simulation analysis the ALHAT team has determined that a single frame of flash lidar data may not be sufficient to build a landing site digital elevation map (DEM) with acceptable spatial resolution, precision and size to meet current system requirements. Similarly, building a mosaic that meets current systems requirements may not be able to be accomplished within the short time window for imaging during a landing.

One way to overcome this potential limitation is by applying SR image enhancement techniques to flash lidar 3-D output images. To date, there are only a few publications that are focused on flash lidar data enhancements.[7–10] Most authors treat flash lidar images as a sequence of uncorrelated images, and do not exploit the real 3-D character of the data. In this paper we describe a new SR algorithm applicable to flash lidar range data that will quickly and effectively create a DEM with sufficient accuracy, precision and size to meet current ALHAT requirements. Our approach is based on the the back projection method. This method has recently been applied to solve the super resolution problem, however this effort was performed in the 2-D environment.[11] We outline the sequence of steps to implement our 3-D data algorithm and analyze the performance of our algorithm by processing data generated from a high fidelity model of a flash lidar that contains parameters for various sensor functions and noise sources. The flash lidar model is embedded in a full end-to-end landing simulation in which it is attached to a gimbal on a spacecraft model that flies varying trajectories to a lunar landing site. The gimbal points the flash lidar to the target landing site throughout the trajectory. During the spacecraft descent, the flash lidar illuminates a high-resolution synthetic lunar elevation map, processes the returned laser light within the field of view of the sensor, and outputs the calculated sensor range data. The range images returned by the model are processed using our SR algorithm and a landing site DEM is generated. Our results show that the achieved level of accuracy and precision of our SR generated DEM is more than adequate for detecting hazardous terrain features and identifying safe areas.

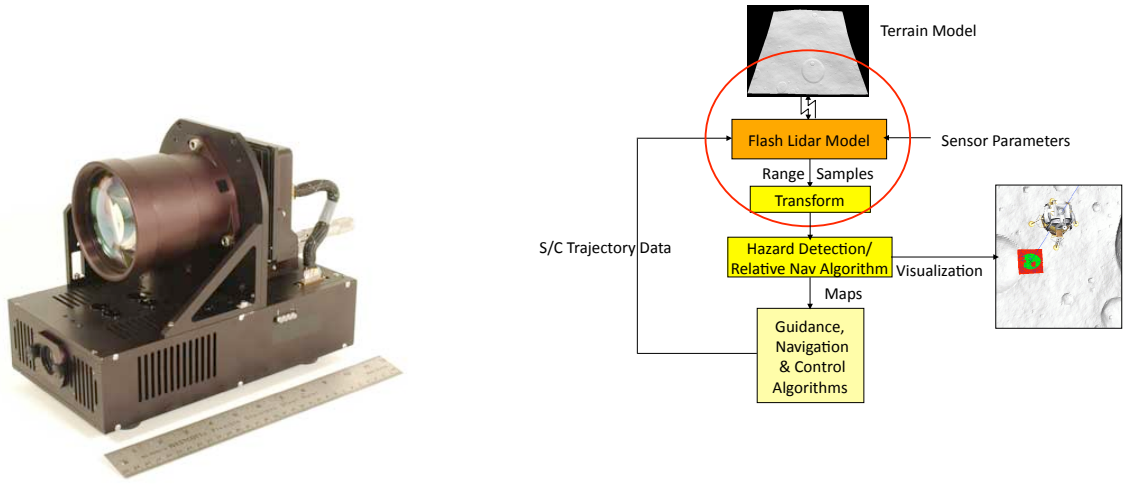
Another problem related to building the DEM that our SR algorithm may help to alleviate is finding the lidar position and orientation in space. Our algorithm is capable of generating the sensor, and subsequently the spacecraft, position to a very high level of accuracy in real-time. This important information may assist with the implementation or operation of the navigation algorithms within the ALHAT project. This capability will be the subject of a future investigation. In this paper we focus solely on our development of a fast and robust method to reconstruct a DEM generated with our SR algorithm. First, we briefly describe a flash lidar and our hi-fidelity model of the system in Section 2. In Section 3 we describe the basis of our algorithm — the back projection method, and outline the procedure to implement our algorithm. In Section 4 we describe our computational experiments and results and in Section 5 we discuss our conclusions.

2. FLASH LIDAR DESCRIPTION

In general, flash lidars transmit a relatively wide divergence, pulsed laser beam to illuminate a target surface. Reflected, and ambient, light in the field of view (FOV) of the sensor is received via an optical subsystem, detected by a focal plane array (FPA), translated into electrical signals by a readout integrated circuit (ROIC), and timed and processed on a per pixel basis to generate time-of-flight range measurements. These range measurements can be interpreted as a 3-D image and transformed into a corresponding DEM. The relative intensities of the returned signals are also measured and can be used to generate an intensity image.

Several different versions of flash lidars are being introduced by companies such as Advanced Scientific Concepts (ASC) and Northrop Grumman (NGC). The ALHAT project is investigating these sensors, and others, for potential use, and it is advancing individual flash lidar components and integrating them into current systems to improve performance. NASA Langley Research Center is managing the acquisition, augmenting the capabilities, and tuning the performance of the sensors for the project. Flash lidars from ASC have been acquired, tested with calibrated targets,[12] tested in the field from an airborne platform,[13] and are currently being analyzed for potential performance improvements. A picture of an ASC flash lidar is shown in Figure 1. This lidar contains a pulsed laser working at 1.5mk wavelengths and a 128×128 FPA. Improvements such as incorporating a higher power laser, operating at more responsive wavelength (1.06 mk), and incorporating a larger 256×256 FPA and ROIC are currently being developed, integrated and tested.

Because of the complexity inherent in a lidar system and the diverse mixture of sensors and components being studied, a physics-based, hi-fidelity model of a flash lidar was developed that can be used to analyze and validate the behavior of different sensors operating under various conditions and in different configurations. The model was targeted for use within the NASA Program to Optimize Spacecraft Trajectories (POST II) simulation environment.[14] POST II is an end-to-end descent and landing simulator that incorporates planetary body gravity models, topography and terrains, and landing vehicles and their respective landing systems including



(a) ASC Flash Lidar

(b) Flash Lidar Model Environment - I/O

Figure 1: Picture of the ASC flash LIDAR and the flash model environment

guidance, navigation, and control (GNC) algorithms, and models of the vehicle sensors such as our flash lidar. The ALHAT project is currently using POST II to perform extensive, detailed spacecraft landing analysis.[15]

A subset of the simulation environment that illustrates the model context and interfaces is shown in Figure 1. The flash lidar model interacts with a hi-resolution simulated lunar terrain via a proprietary mapping and ray-trace routine. Multiple rays are assigned to each pixel within the FPA of the model. The ray-trace routine is used to interrogate the lunar terrain within the ground footprint of the sensor. A ground footprint is generated for both the transmitter and receiver from several parameters such as the laser beam divergence, receiver FOV, and the model's attitude and position which are derived from the spacecraft slant range, path angle and gimbal orientation. Range and surface reflectivity data for each ray are returned to the core processing code of the model. Based on this input information, the model then implements a standard convolution imaging equation:

$$\zeta(x,y,t) \quad = \quad \rho(x,y,t) \otimes \vartheta(x,y,t) + \eta(t) \tag{1}$$

where ζ is the output signal, ρ is the point spread function that represents the lidar temporal pulse profile, ϑ is the object function that describes how the transmitter energy is scattered in amplitude and time, and η is the total noise sum of various random processes such as detector dark noise, Johnson noise and shot noise. Range information is generated on a per pixel basis by signal reconstruction algorithms. For ALHAT, the range data is then transformed into an angle-angle-range format and transferred to either a hazard detection or hazard relative navigation algorithm. These algorithms produce maps that are fed into a GNC algorithm which controls the motion of the spacecraft. Further details on the internal mathematics of the flash model will be discussed in a future publication article.[16]

The flash lidar model has several parameters that can be adjusted to enable simulation of a wide variety of sensor specifications. Example parameters include instantaneous field of view (IFOV), receiver aperture, laser power, beam divergence and sample rate. The parameters are loaded from an i-load file into the model before a simulation run and can be adjusted to enable Monte Carlo type simulation. For this study, most of the parameters have been tuned to simulate the functionality and operation of the ASC sensor.

The following steps are performed to setup and generate data to test our super-resolution algorithm. First, a spacecraft trajectory is generated for 90° and 45° path angles with an initial slant range of 1000 m and a trajectory that descends linearly down to 100 m. We made the following assumptions: the spacecraft moves

Slant Range (m)	IFOV (rads)	Receiver Aperture (rads)	Transmitter Beam Divergence (rads)
1000	0.00040	0.05	0.0512
750	0.00053	0.0375	0.0678
500	0.00080	0.025	0.1024
250	0.00160	0.0125	0.2048

Table 1: Model parameters adjusted during the trajectory to simulate zoom optics

from the beginning to the end of the trajectory in 30 seconds and the frame rate of the lidar is 20 Hz, so a new image is taken every 1.5 m of descent. We also programmed the gimbal to point at the target landing site throughout the trajectory except where we inject random noise on specific axes to test our algorithm. Second, an i-load file for the model is created with baseline sensor parameters. For example the FPA is set to 128×128 pixels and the range precision is set to 5 cm. The IFOV, receiver aperture, and transmitter beam divergence are adjusted during the descent as shown in Table 1 to simulate zoom optics adjustments that may be made to keep roughly the same ground footprint throughout the trajectory. Third, a synthetic lunar map with 10 cm posting is loaded into the simulation as the representative landing site target for the lidar. The map has an area of approximately 450×450 m, and minimum and maximum map height data of -9.4 m and 4.87 m respectively. After setup is complete, the simulation is executed and range output images are generated for each trigger point throughout the trajectory.

3. BACK PROJECTION METHOD

The term back projection is commonly used in the area of linear X-ray tomography in which the measured signals are 1D or 2-D projections of 2-D or 3-D objects, respectively. Calculation of the projections, i.e. solving the direct problem, is performed by integrating an object quantity feature, for example density, along the X-ray path. Back projection, i.e. solving the inverse problem, occurs via a reconstruction technique where the property of projections are translated back into the object following the X-ray path in the inverse direction. This concept can also be applied to flash lidar data. Using the back projection method for flash lidar data implies a model of the measuring process where the following is valid: the range recorded by a detector is the average of the ranges from the detector location to the each point on the reflecting surface which is located inside of the detector FOV as described by the following equation:

$$R_i(x_1, x_2) \quad = \quad 1/S_i \times \iint\limits_{\Omega_i} R(x, y)dx \tag{2}$$

where R_i is the range recorded by detector number i; $R(x, y)$ is the range from the point with coordinates (x, y) on the surface, S_i, to the detector; and Ω_i is the domain on the surface corresponding to the detector FOV. In fact, Ω_i is the projection of the FOV on the flat plane, and S_i is the area of this domain as shown in Figure 2. This model is an approximation, and the level of accuracy of this approximation can be tested only after the inverse problem is solved.

The purpose of our SR method is to determine elevations on a mesh with cells smaller than Ω. If Θ is defined as a fine rectangular mesh with pixel sizes that meet our specified requirement, then Equation 2 can be rewritten as

$$R_i \quad = \quad 1/N_i\big(\sum_{k=1}^{k=N_i} R_k\big) \tag{3}$$

where k denotes the number of cells on the mesh Θ that overlap with the projection Ω_i, and N_k is the number of such cells. Equation 3 is a first approximation to Equation 2. Weighting factors could be included to increase the accuracy of the formula, The geometry of the back projection procedure is illustrated in Figure 3. The algorithm can be executed by performing the following steps:

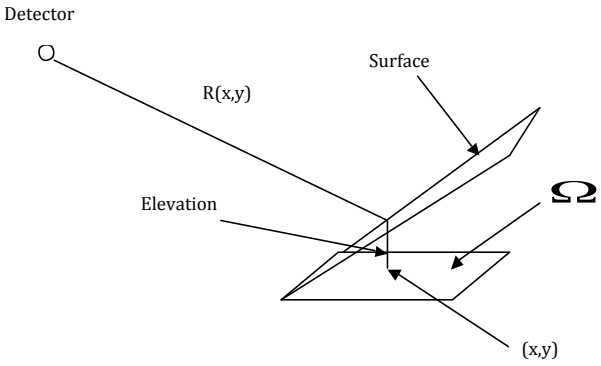


Figure 2: Projection of the FOV on the flat plane for one detector

1. The SR elevation map is initialized as the fine mesh Θ. Create arrays S, M and D. Arrays $S(k)$ and $M(k)$ are working arrays with dimensions the same as Θ. Index k is a 3-D index into the fine mesh Θ. Array $D(k)$ will contain the algorithm output — the elevation map on the fine mesh. All arrays are initialized to zero.

2. For each frame and for each individual detector of the FPA, determine the projection of the FOV on the reference surface (zero elevation level).

3. Find the set of cells from Θ that overlap with this projection. Using the center of the cell from Θ as one point, connect this point to the detector by a straight line and calculate the distance.

4. The difference between this distance and the range registered by the detector indicates the location of the point where the ray crosses the surface. The elevation of this point is calculated by the formula:

$$h_{ki} \quad = \quad (R_k - R_i) \times \sin\theta_i \tag{4}$$

   Points k and k_1 may be the same or different depending on the angle of incidence.

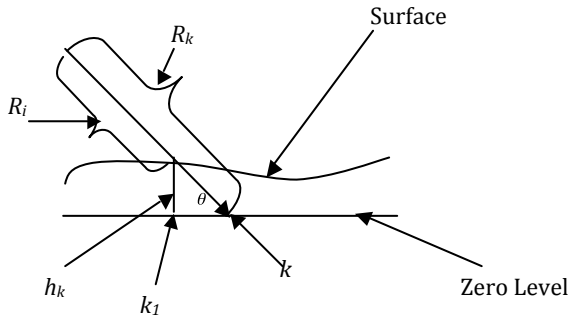


Figure 3: Geometry of the Back Projection algorithm

5. Update arrays S and M as

$$\begin{aligned} S(k_1) &= S(k_1) + h_{k1} \\ M(k_1) &= M(k_1) + 1 \end{aligned} \tag{5}$$

6. Continue this operation for all detectors in the current frame. Then apply the following normalization procedure:

$$Dk = S(k)/M(k) \text{ for all } k \tag{6}$$

7. Repeat steps two to six for all frames.

4. COMPUTATIONAL EXPERIMENTS

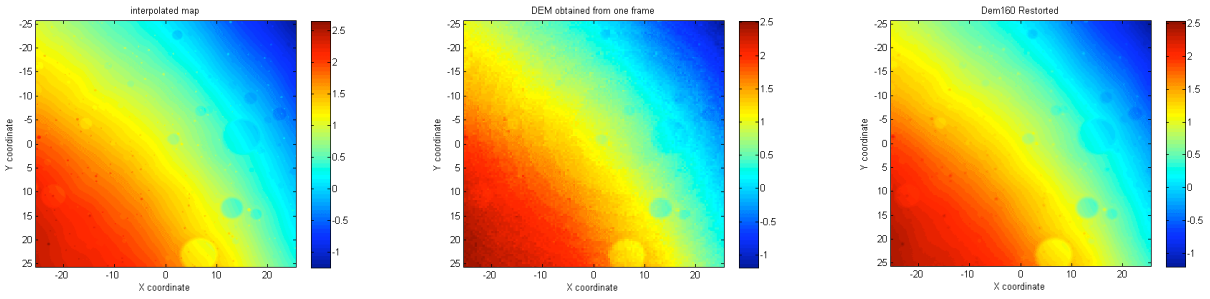
Several computational experiments were performed to test our SR procedure. These experiments focused primarily on construction of a DEM with sub-pixel resolution using data collected from various trajectories. All tests were performed using five basic steps:

1. Generate a simulated spacecraft trajectory and select a target map
2. Simulate the generation of flash lidar data (frames) throughout the trajectory
3. Add controlled random noise to all frames if testing under these conditions
4. Apply the back projection method and construct the restored DEM
5. Compare the original "true" frames with the constructed DEMs using basic statistical metrics.

4.1. Test Case 1

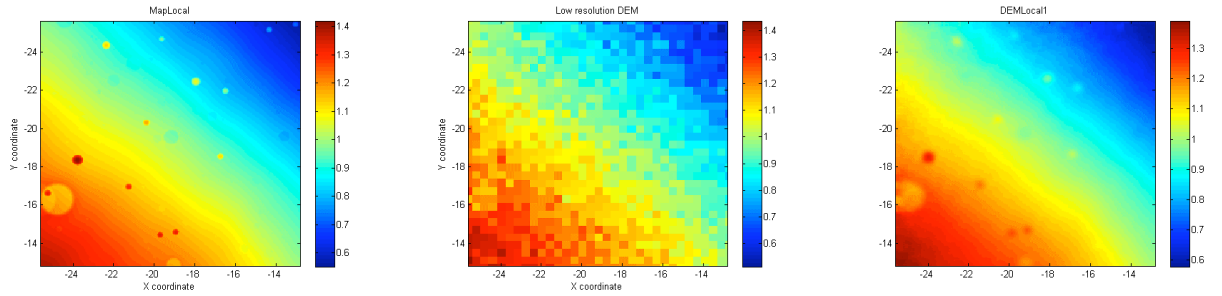
For this test case, and all subsequent test cases, the model of the flash lidar described in Section 2 is used to generate the simulated lidar data. The original map was chosen from a synthetic lunar terrain database in POST II. The map is classified as a smooth mare terrain type with 10% rock abundance. The lidar input map, shown in Figure 4a, was created from a portion of the original lunar map on a rectangular grid with 1024×1024 pixels. Lidar data were generated using a nadir (90°) approach trajectory to this map starting with a slant range of 1000 m descending down to 100 m. The size of the frames were 128×128 pixels. Zoom optics were implemented as previously discussed with zoom changes as shown in Table 1. A low resolution DEM was generated, shown in Figure 4b, using only the first frame.

Our back projection algorithm was then applied using the first 160 frames. The results are shown in Figure 4c. To better observe the visual quality of the maps, "magnified" images of the input, low resolution



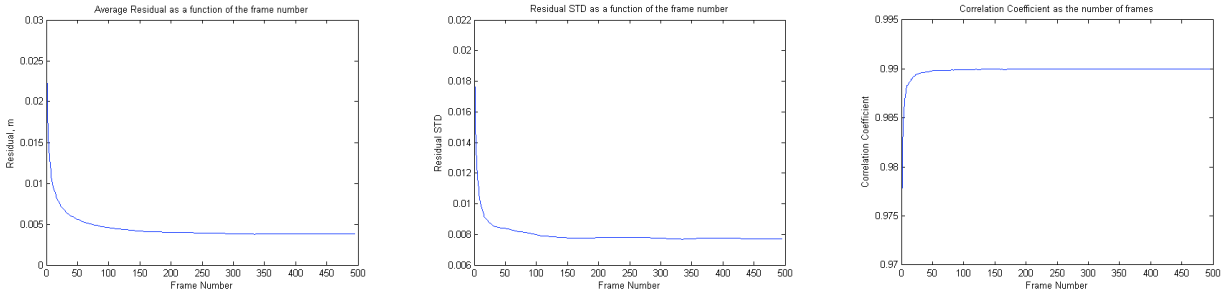
(a) Lidar input elevation map (b) DEM generated using only the first frame (c) Restored DEM using 160 frames

Figure 4: Input map, DEM generated using one frame and DEM generated from 160 frames



(a) Original Map (b) Restored low resolution map (c) Restored high resolution map

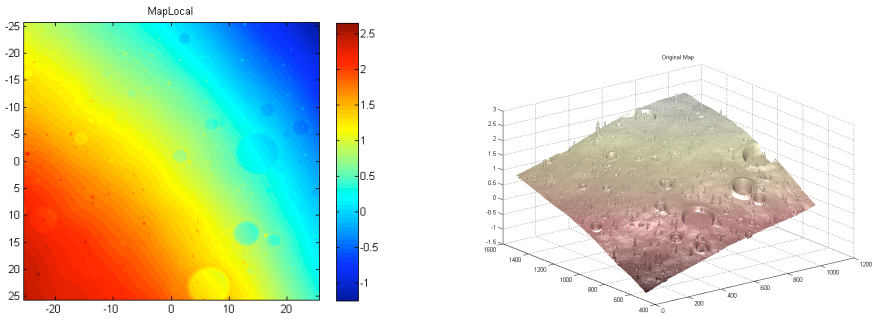
Figure 5: Comparison between original, high and low resolution maps for part of the image.



(a) Average residual (b) Residual STD (c) Correlation coefficient

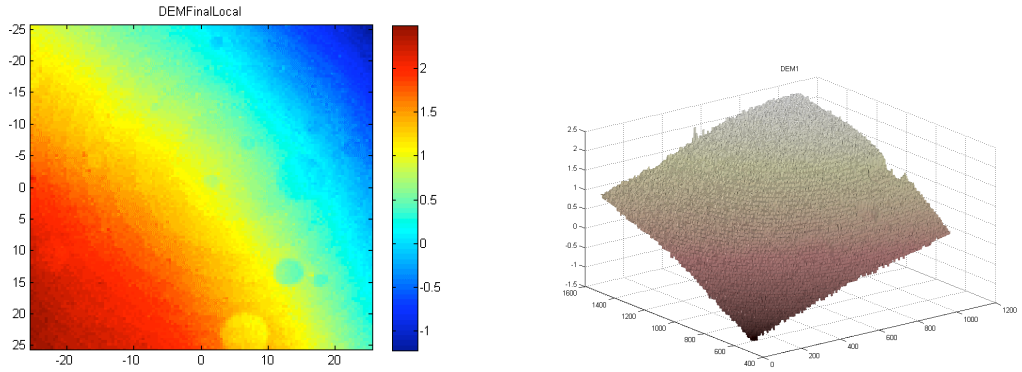
Figure 6. Mean of absolute value of residual, standard deviation of residual, and correlation coefficient between original and restored map as a function of frame number.

and high resolution maps are presented. The maps were taken from a using a 256 × 256 pixel subset of the full maps. These maps are presented in Figure 5. It is obvious that the restored high resolution map is not perfect, but it shows dramatically improved image quality. This can also be seen when observing the comparative image statistics. The results of changes in the mean residual, residual STD, and correlation coefficients for the maps are shown in Figure 6. Minimal increases in image quality are gained after only a few frames. To see this convergence after 30 frames we now use 2-D and 3-D representations which show the details in the maps more clearly. The original map is shown in Figure 7 and the results of increasing the iteration interval from 1 to 30 frames are shown in Figures 8 and 9, respectively. The improvement in image quality can clearly be seen.



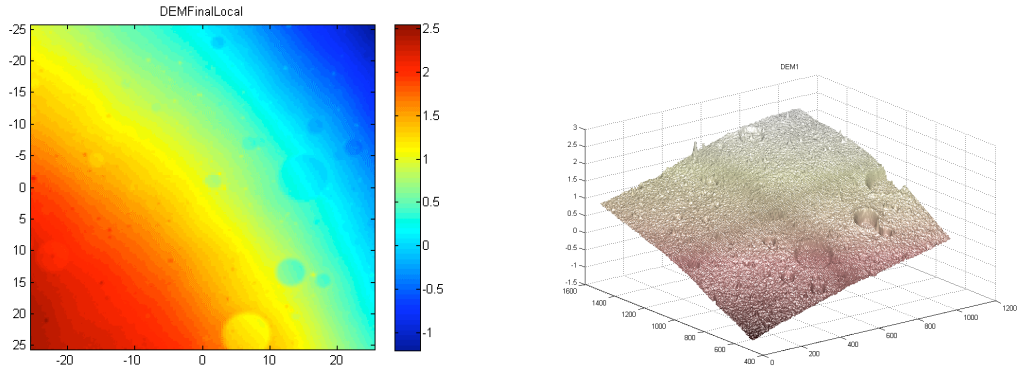
(a) Restored map 2-D (b) Restored map 3-D

Figure 7: Original map in color coded 2-D and 3-D representations



(a) Restored map 2-D (b) Restored map 3-D

Figure 8: Restored map using 1 frame



(a) Restored map 2-D (b) Restored map 3-D

Figure 9: Restored map using 30 frames

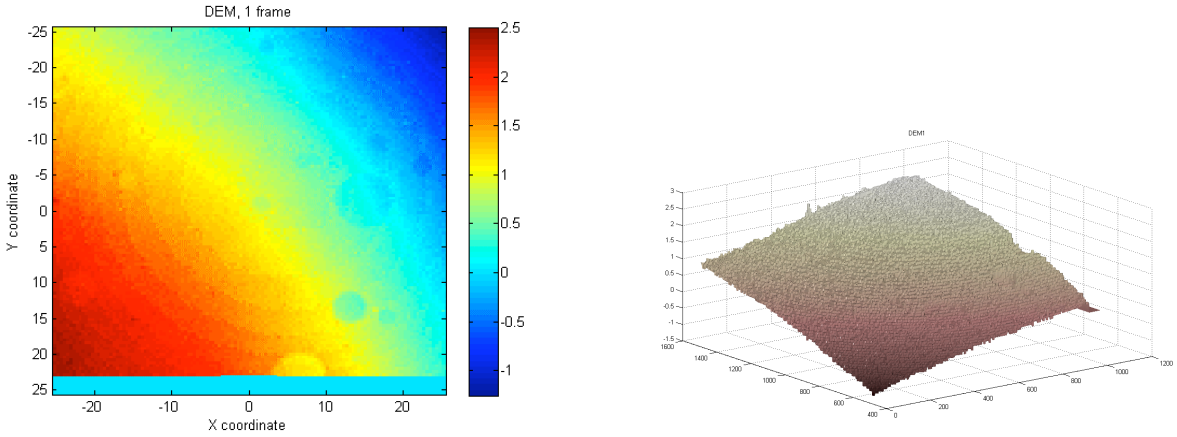
4.2. Test Case 2

For this test case the synthetic flash lidar data were generated using the same set of lidar parameters, optical zoom changes, and map that were used in Section 4.1. The spacecraft trajectory is also the same except that simulated jitter has been injected onto the spacecraft by applying a normally distributed random variable with zero mean and a standard deviation of 0.1° (1.745 mrads) along the spacecraft z-axis. The results of increasing the iteration interval from 1 and 30 frames are shown in Figures 10 and 11, respectively.

The mean of the residual, standard deviation of residual, and correlation coefficient, all calculated between the true and restored maps are shown on Figure 12. The results indicate a slight improvement in the performance of our SR algorithm when jitter is applied to the flash lidar, and thus subsequent output data, than without jitter. This is due to the inherent nature of SR processing which takes advantage of any slight misalignment in correlated images. Thus, this type motion may be used to our advantage since any real spacecraft will have some level of jitter during operation.

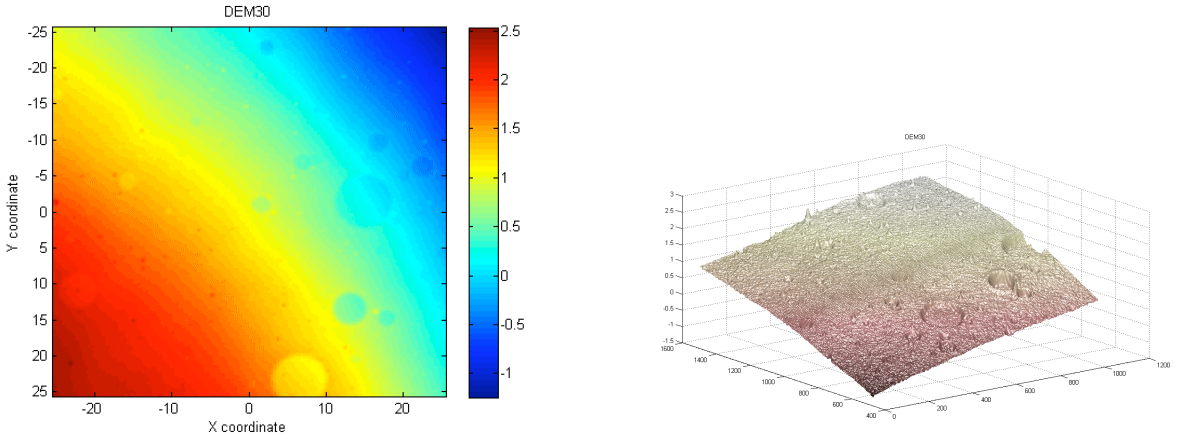
4.3. Test Case 3

For this test case the synthetic flash lidar data was generated using the same set of lidar parameters, optical zoom changes, and map that were used in Section 4.1. Simulated jitter has been removed and a new, initial 45° incidence angle trajectory was flown by the simulated spacecraft. The spacecraft moves from $\approx$ 700 m down to 250 m in altitude (z-axis), while the slant range decreases from $\approx$ 1000 m down to 100 m.



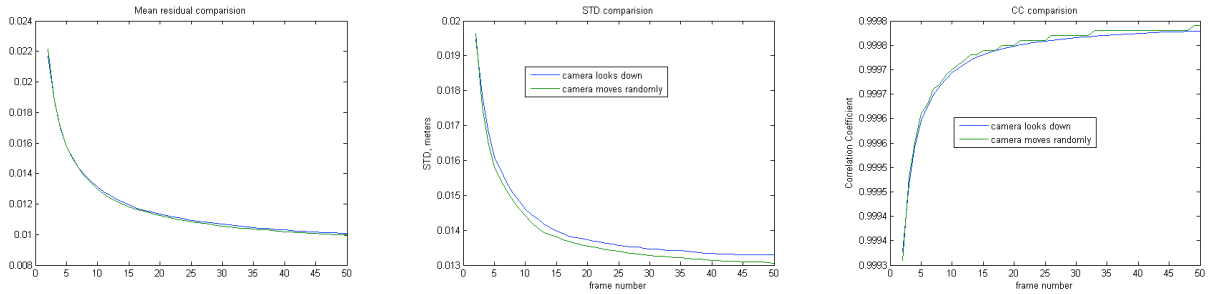
(a) Restored map 2-D (b) Restored map 3-D

Figure 10: Lidar moving restored map using 1 frame



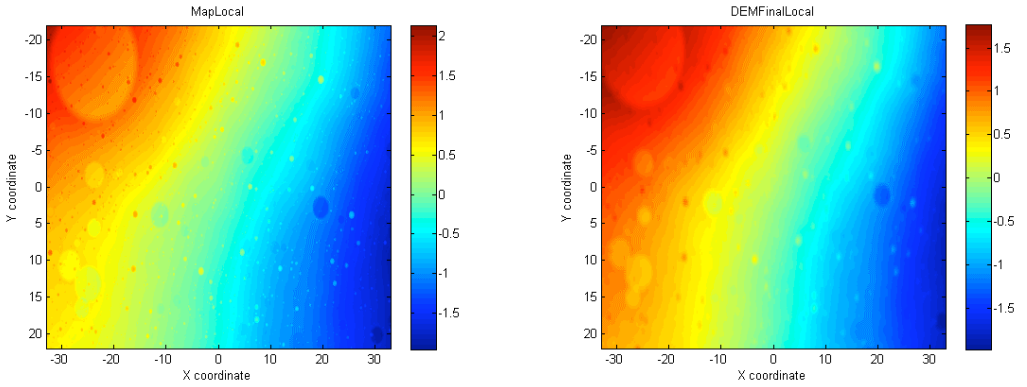
(a) Restored map 2-D (b) Restored map 3-D

Figure 11: Lidar moving restored map using 30 frames

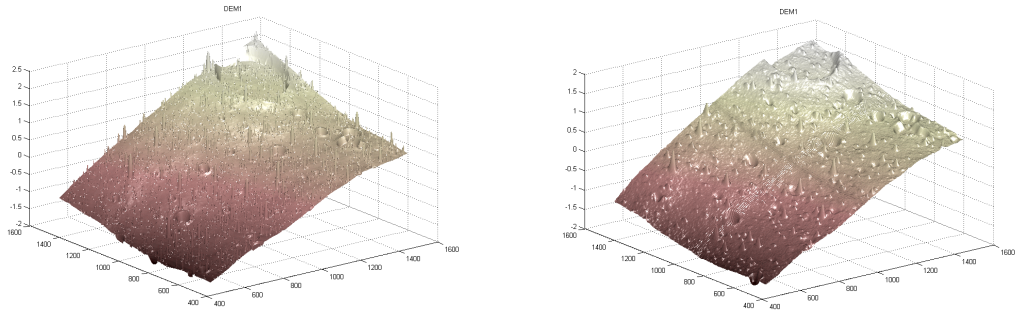


(a) Mean residual (b) STD comparison (c) Correlation coefficient comparison

Figure 12. Mean of absolute value of residual, standard deviation of residual, and correlation coefficient between original and restored map for frames 1- 50 calculated for two cases.

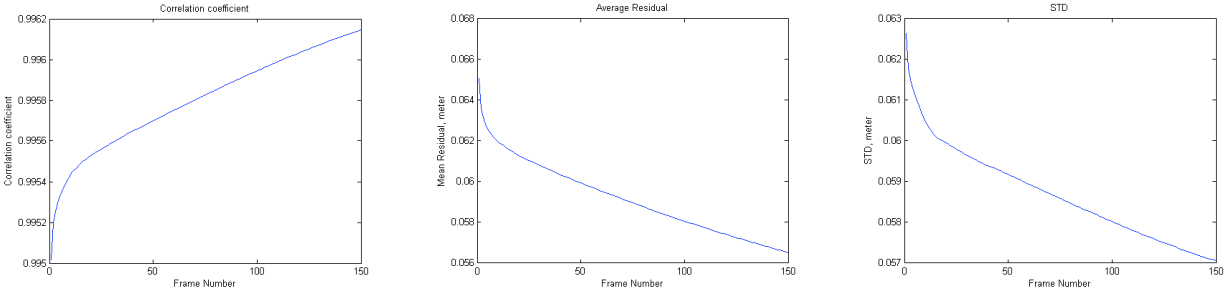


(a) Original map 2-D (b) Restored map 2-D



(c) Original map 3-D (d) Restored map 3-D

Figure 13: Original and restored elevation maps in 2-D and 3-D representations



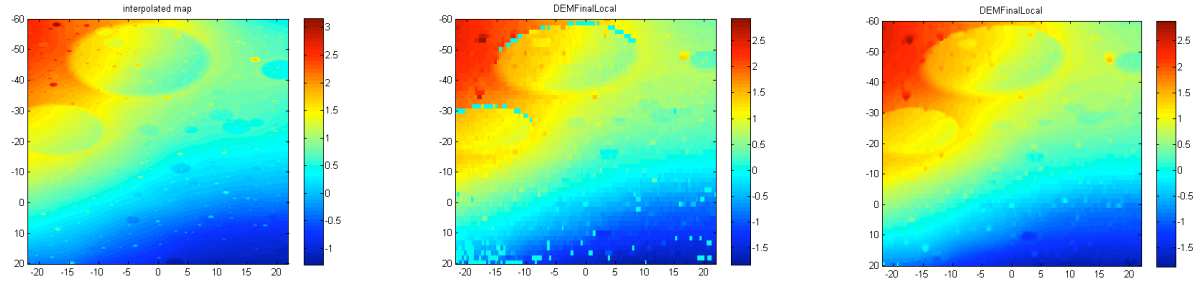
(a) Correlation coefficient (b) Average residual (c) STD

Figure 14. Correlation coefficient, mean of absolute value of residual, and standard deviation of residual between original and restored map for frames 1- 150.

The original map and the results of processing with our SR algorithm are presented in Figure 13 in both 2-D and 3-D representations. The iteration interval used for processing was 50 frames. The correlation coefficient, mean residual, and standard deviation between the original and restored images are shown in Figure 14. The initial noise level on the ranges was $\approx$ 10 cm and the measurements contained a significant number of dropouts. After SR processing, the standard deviation of the residual between the original "true" map and our restored map is improved to $\approx$ 5cm.

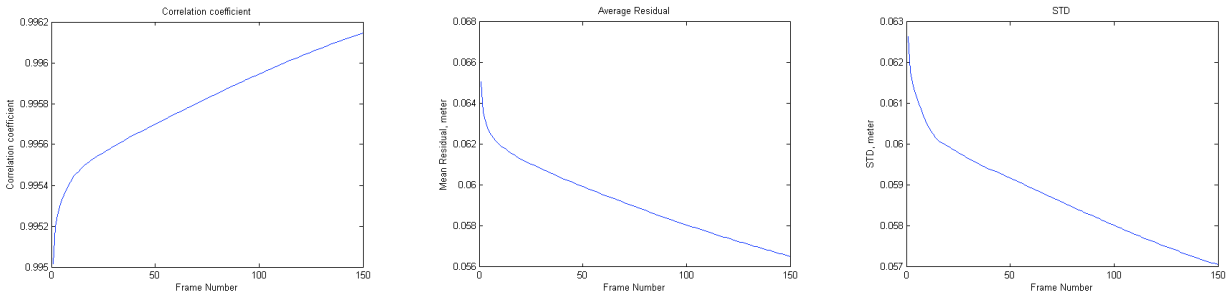
4.4. Test Case 4

In this test case, the synthetic flash lidar data was generated using the same set of lidar parameters, optical zoom changes, and map that were used in Section 4.2. The trajectory flown by the simulated spacecraft has been modified to have a challenging 15° incidence angle. The original input map is shown in Figure 15a. Again, our SR algorithm was only applied to the areas that were covered by the lidar footprint. Our restored DEM using iteration intervals of 1 and 50 frames are shown in Figures 15b and c respectively. Again, the increase in image quality can easily be seen when we process multiple frames. Even for this shallow angle trajectory, our SR algorithm provides significant improvement. The correlation coefficient, mean residual, and standard deviation between original and restored images are shown in Figure 16. These statistics corroborate our previous results.



(a) Original map (b) Restored map - one frame (c) Restored map - 50 frames

Figure 15: Original and restored elevation maps.



(a) Correlation coefficient (b) Average residual (c) STD

Figure 16. Correlation coefficient mean of absolute value of residual, and standard deviation of residual between original and restored map for frames 1- 150.

5. CONCLUSIONS

In this paper, we presented a new super-resolution algorithm that is applicable to flash lidar range data. We described the basis of our algorithm, the back projection method, and outlined the computational steps to perform the algorithm. We used a hi-fidelity model of a flash lidar to produce input range images to test our algorithm. The flash lidar model was attached to a gimbal on a spacecraft model flying various landing trajectories. The gimbal pointed the lidar to a target landing site throughout the trajectory. The flash lidar illuminated a synthetic lunar terrain, processed the returned signal with typical sensor noise characteristics, and output a range image which was subsequently processed by our SR algorithm. Original, input maps to the lidar, and our corresponding SR processed maps were presented for visual quality and compared with basic statistical measures.

The results show that our proposed algorithm is capable of improving the spatial resolution of noisy, low resolution flash lidar images, and providing a high level of accuracy and precision in the final output DEMs. Our algorithm is also robust to different viewing angles. Another important advantage of our algorithm is that it has the potential to be performed in real-time using current commercial-off-the shelf hardware processors. This production of high quality imagery and potential for real-time implementation shows significant promise for use within the ALHAT system. Combined together with an ortho-rectification algorithm, and a hazard detection/hazard relative navigation algorithm, our SR algorithm can be used to provide a critical component for performing autonomous and safe landings.

6. ACKNOWLEDGMENTS

The authors thank the NASA Exploration Technology Development Program Office, the Autonomous Landing and Hazard Avoidance Technologies project, and the NASA LaRC Systems Engineering Directorate for the funding and support that made this work possible. The authors also thank the management team at AMA Inc. for their continued support.

REFERENCES

1. S. Park, M. K. Park, and M. Kang, "Superresolution image reconstruction: A technical review," *IEEE Signal processing Magazine* **20**, pp. 21–36, May 2003.
2. S. Farsiu, M. D. Robinson, M. Eland, and P. Malinfar, "Fast and robust multiframe superresolution," *IEEE Transactions on Image Processing* **13**, pp. 1327–1344, 2004.
3. S. S. Young and R. Driggers, "Superresolution image reconstruction from a sequence of aliased imagery," *Appied Optics* **45**, pp. 5073–5085, 2006.
4. G. Clement, J. Huttunen, and K. Hynynen, "Super-resolution image reconstruction from a sequence of aliased imagery," *Journal of the Acoustical Society of America* **118**, pp. 3953–3960, 2005.
5. S. Chaundhuri, *Super-resolution Imaging*, Springer, 2001.
6. C. Epp, E. Robinson, and T. Brady, "Autonomous landing and hazard avoidance technology (ALHAT)," in *IEEE Aerospace Conference*, pp. 1–7, 2008.
7. Q. Yang, R. Yang, J. Davis, and D. Nister, "Spatial-depth super resolution for range images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
8. G. Rosenbush, T. Hong, and R. Eastman, "Super-resolution enhancement of flash LADAR range data," in *Proceedings of SPIE*, **6736**, 2007.
9. S. Hu, S. S. Young, T. Hong, J. Reynolds, K. Krapels, B. Miller, J. Thomas, and O. Nguyen, "Super-resolution for flash LADAR data," in *Proceedings of SPIE*, **7300**, 2009.
10. S. Hu, S. S. Young, T. Hong, J. Reynolds, K. Krapels, B. Miller, J. Thomas, and O. Nguyen, "Super-resolution for flash LADAR data," *Applied Optics* **49**(5), pp. 772–780, 2010.
11. F. Qin, X. He, W. Chen, X. Yang, and W. Wu, "Video superresolution reconstruction based on subpixel registration and iterative back projection," *Journal oF Electronic Imaging* **18**(1), p. 1300, 2009.
12. D. Pierrottet, F. Amzajerdian, B. Meadows, R. Estes, and A. Noe, "Characterization of 3-d imaging lidar for hazard avoidance and autonomous landing on the moon," in *Proceedings of SPIE*, **6550**, 2007.
13. A. Bulyshev, D. Pierrottet, F. Amzajerdian, G. Busch, M. Vanek, and R. Reisse, "Processing of 3-dimensional flash LIDAR terrain images generated from an airborne platform," in *Proceedings of SPIE*, 2009.
14. G. Bauer, D. Cornick, and R. Stevenson, "Capabilities and applications of the program to optimize simulated trajectories (post)," in *NASA CR-2770*, February 1977.
15. J. Davis, S. Striepe, R. Maddock, G. Hines, S. Paschall, B. Cohanim, T. Fill, M. Johnson, B. R., K. DeMars, R. Sostaric, and A. Johnson, "Advances in POST2 end-to-end descent and landing simulation for the ALHAT project," in *AIAA/AAS Astrodynamics Specialist Conference, AIAA-2008-6938*, 2008.
16. G. Hines and et al., "Hi-fidelity modeling of a 3-d flash lidar," in *unpublished manuscript*, 2010.