LuVo: Lunar Visual Odometry using Homography-based Image Feature Matching

Ryan Soussan[1,2,3], John McCaffery[1,2], Scott McMichael[1,2], Matthew Deans[1]

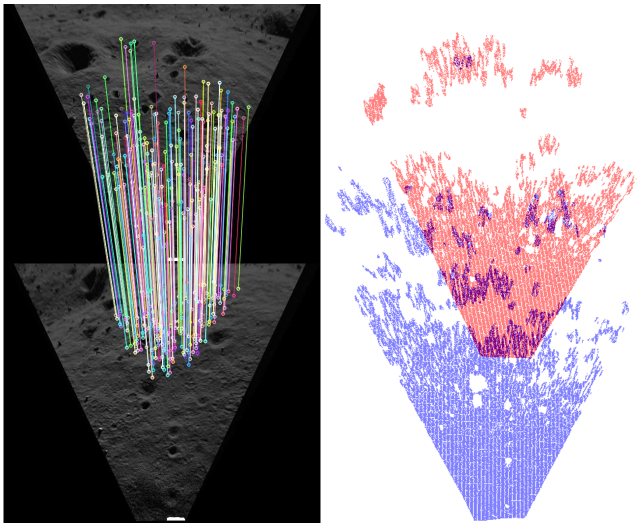
***Abstract*— We present LuVo, an initialization-free stereo visual odometry (VO) method developed for the VIPER lunar rover. We provide a novel stereo registration method using LightGlue image feature matching in a warped, locally planar space that improves matching robustness to larger baseline stereo sequences and repetitive terrain that traditionally challenge odometry approaches. We additionally introduce methods that increase the usable image region for matching by estimating a horizon cutoff in image space and enhance robustness to stereo correspondence failures using a Manhattan distance search for valid stereo points during cloud alignment. We evaluate the performance of LuVo on a dataset of 155 simulated lunar stereo sequences and show that it significantly improves registration accuracy and success rates for clouds separated by both expected driving ranges below eight meters and longer distance translations of up to 16 meters. While LuVo is developed for VIPER, it can be used in other environments featuring slip-prone and repetitive terrain that limit rover travel.**

I. INTRODUCTION

The VIPER lunar rover is designed to explore the south pole of the Moon in search of water ice [1]. It navigates using onboard wheel and inertial odometry, stereo cameras, and front-facing lights, as shown in Fig. 2. VIPER performs global pose estimation by matching panoramas to Digital Elevation Models (DEMs) [2], but accuracy is limited to meter-scale due to the low-resolution of the satellite images used for DEM construction. Additionally, this is only done every 50-100 meters to limit travel delay. To keep navigation errors under three meters for every 224 meters of driving as desired [1], reliable relative pose estimation is needed.

Lunar and Martian rovers use combinations of point cloud alignment and feature tracking for relative localization [3]. Alignment methods, such as iterative closest point (ICP) [4] and point-to-plane ICP [5], rely on low-noise pose initialization. Similarly, feature tracking approaches require either accurate pose initialization or large amounts of image overlap [6], [7]. However, to conserve limited solar energy, VIPER disables its cameras and lighting during drives and only captures images every five to eight meters. These large translations subject pose initialization using onboard wheel odometry to accumulated drift due to wheel slip [8]. They also limit image overlap for feature tracking, forcing VIPER to seek alternative solutions for visual odometry.

Machine learning feature matchers, such as SuperGlue [9] and LightGlue [10], increase matching robustness in redundant environments like the lunar surface. Since VIPER and other recent rovers use ground-based servers [1], [11] or landers [12], [13] for additional computation, these matchers can be utilized for navigation. However, they still struggle to handle significant scene changes rendered by long distance drives. Akin to Earth-based robots using geometric assumptions such as the Manhattan World [14], VIPER and other wheeled robots can address this by leveraging the relatively flat terrain of the Moon and Mars, featuring few non-planar obstacles. Navigation images can therefore be projected to the ground plane, improving the visibility of recurring features when faced with sizable image separations.



(a) LuVo registration.

(b) Cloud overlap.

Fig. 1. LuVo successfully registers stereo pairs separated by 8.3 meters on the lunar surface using homography-based LightGlue image matching shown in Fig. 1a. Here, warped-perspective left stereo images are displayed with estimated matches drawn using colored lines. ICP, however, fails to register the stereo clouds for the same data, illustrated in Fig. 1b. Sparsity in the far range of the reference cloud in blue limits overlap for alignment during longer distance drives.

We therefore present LuVo, an initialization-free lunar stereo visual odometry method robust to large relative translations in slip-prone and repetitive environments. Our contributions include:

- Image feature matching using LightGlue in a surface-aligned space to improve matching success for large baseline and repetitive images.
- Horizon row detection using valid stereo points to increase the usable projected image area and further boost matching performance.

The authors are with [1]NASA Ames Research Center, Moffett Field, CA, 94035, USA, [2]KBR, Inc, and [3]Aerodyne Industries. {ryan.soussan, john.mccaffery, scott.t.mcmichael, matthew.deans}@nasa.gov

- A Manhattan distance search for valid stereo points in sparse regions of clouds to enhance registration reliability.
- An analysis of ICP performance using different noise profiles for predicted drive distances.

We evaluate LuVo on a dataset of 155 simulated lunar stereo sequences and show that it exhibits significant improvements in accuracy and success rates, both for expected driving ranges and drives of up to 16 meters, compared to image space matching and ICP-based methods. LuVo is designed for lunar rovers, but can be used in other environments where long relative translations and slip-prone or repetitive terrain challenge visual odometry approaches.

II. Related Work

A. Moon

The Yutu-2 lunar rover [15], [11] for the Chang'e 4 mission fuses inertial measurements with both SURF and manually selected features using a bundle adjustment-based pipeline. It captures stereo images in a panorama at various yaw and pitch angles between drives, and later selects stereo sequences containing the most overlap for relative pose estimation. While VIPER records panoramic images every 50-100 meters for DEM-based global localization, it uses stereo sequences taken at single viewpoints, spaced five to eight meters apart, for relative navigation. This enables faster operation, but prevents overlap from occurring beyond sequential images as needed for persistent feature tracking.

The CADRE rovers [13] perform keyframe-based Kalman Filter navigation by combining image features with IMU and sun sensor readings. Concurrently, their lander tracks the rovers using Ultrawide-band (UWB) sensors. Similarly, the Pragyan rover [12] for Chandrayaan-3 employs an unspecified stereo alignment method for navigation and relies on its lander for additional stereo calculations. Keyframe-based methods and stereo alignment both rely on repeated feature visibility or low-noise pose initialization, which are not available to VIPER due to its long-distance driving.

Wagner *et al.* [16] add Kanade-Lucas-Tomasi (KLT) [17] feature tracks in a Kalman filter for the CSA Artemis project, while Li *et al.* [18] use Harris corners and an image intensity cost for feature matching. Both of these approaches, however, assume small-baseline travel or high-precision initialization.

LunarNav [19] and ShadowNav [20] use stereo cameras and LiDAR to detect craters and align them to crater landmarks in orbital maps, but accuracy is limited to several meters due to low map resolution. Chelmins *et al.* [21] propose using radiometric ranging from Lunar Relay Satellites for localization, but this yields an accuracy of only ∼1 meter, even after five minutes of processing.

B. Mars

Mars rovers, from Spirit and Opportunity [3] to Perseverance [22], perform visual odometry on a 20 MHz CPU using Harris corner matches between stereo clouds. However, they initialize stereo alignment with wheel odometry, which limits driving distances in unconsolidated terrain to ensure alignment begins near a local minimum.

The Ingenuity helicopter [23] uses a downward facing camera to acquire KLT feature tracks. It fuses these with IMU and LiDAR altimeter measurements using an Augmented Kalman filter, but processes images at 30 Hz, providing much smaller relative translations than VIPER.

C. Earth

Similar to lunar and Martian approaches, other stereo and RGB-D VO methods employ combinations of indirect feature matching, direct feature matching, and cloud alignment.

1) Stereo: Indirect methods such as ORBSLAM-ORBSLAM3 [24], [25], [7] use ORB feature matching [26] along with the DBoW2 bag-of-words library [27] to perform simultaneous localization and mapping (SLAM), while OKVIS2 [28], [29] uses BRISK features [30] for keyframe-based SLAM. These methods, however, require repeated viewing of features for accurate tracking and loop closures.

SVO [31], DSO [6], and Basalt [32] use direct costs for image matching, and VINS-Mono and VINS-Fusion [33] employ a combination of the two. Similarly, Kimera [34] and Kimera2 [35] use KLT tracks paired with smart factors [36] for relative pose estimation. Direct methods rely on closely spaced images for reliable tracking and are therefore not suitable for VIPER.

Guan *et al.* [37] and Saurer *et al.* [38] use ground plane and weak Manhattan world assumptions to simplify relative pose estimation, but detect and match features in image space. LuVo matches images in projected space on the locally planar lunar surface to improve reliability.

2) RGB-D: Kinectfusion [39] uses ICP to perform point cloud alignment, while ElasticFusion [40] minimizes point-to-plane errors with direct photometric costs and Zhang *et al.* [41] uses KLT feature tracks [42]. However, each of these methods rely on high-fidelity initialization or small-baseline image sequences.

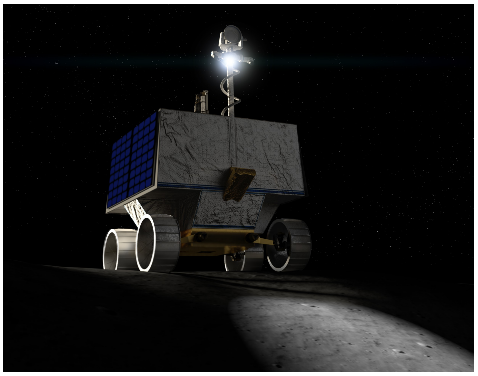


Fig. 2. Rendering of the VIPER lunar rover on the Moon.

III. System Overview

LuVo estimates relative poses between sequential stereo pairs separated by several meters in distance. An overview

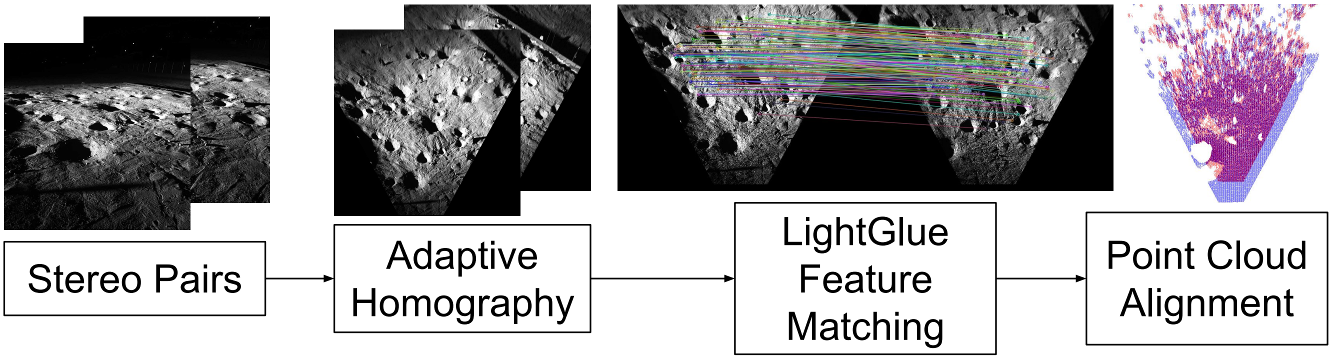
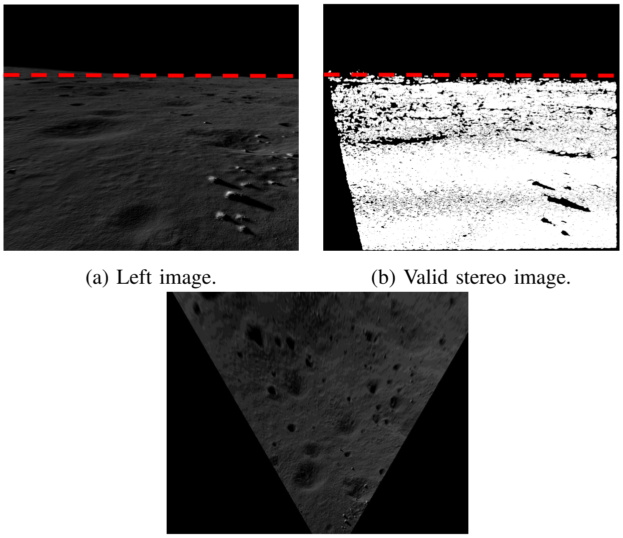


Fig. 3. LuVo uses sequential stereo pairs and applies an adaptive homography procedure to project left images to a locally planar surface for matching. It then computes feature matches using LightGlue before aligning the stereo clouds using nearby valid stereo points for each set of correspondences.


(a) Left image.

(b) Valid stereo image.

(c) Projected homography image.

Fig. 4. Estimated horizon cutoff row shown as a red dotted line in the left stereo image in Fig. 4a. This is calculated using valid stereo points, depicted as white pixels, in Fig. 4b. Removing the region above the horizon increases the usable lunar surface region in the projected image, shown in Fig. 4c, for subsequent image feature matching.

of the pipeline is illustrated in Fig. 3. LuVo's adaptive homography procedure first projects left stereo images onto the planar lunar surface to improve feature matching, described in more detail in §IV. It then detects feature matches between sequential images in warped space using LightGlue, as shown in Fig. 1 and explained in §V. Finally, it aligns clouds using nearby valid stereo points, detailed further in §VI.

IV. Adaptive Homography

The adaptive homography procedure displayed in Fig. 3 generates a surface-aligned image for feature matching. As demonstrated in Fig. 5, matching in warped space improves feature matching for images separated by large translations compared to matching in image space. The adaptive nature of the process stems from expanding the usable area of the projected image by detecting the horizon cutoff, as described in the following sections.

A. Initial Homography

The procedure first estimates the rotation from the camera to ground plane ${}^{\mathrm{G}}_{\mathrm{C}}\mathbf{R}$ per (1):

$${}^{\mathrm{G}}_{\mathrm{C}}\mathbf{R} = {}^{\mathrm{B}}_{\mathrm{C}}\mathbf{R}_{\gamma}\,{}^{\mathrm{B}}_{\mathrm{C}}\mathbf{R}_{\beta} \tag{1}$$

Here ${}^{\mathrm{B}}_{\mathrm{C}}\mathbf{R}$ is the rotation from the camera frame to the body frame, which is locally co-planar with the ground plane, and ${}^{\mathrm{B}}_{\mathrm{C}}\mathbf{R}_{\gamma}$ and ${}^{\mathrm{B}}_{\mathrm{C}}\mathbf{R}_{\beta}$ are the pitch and roll components of ${}^{\mathrm{B}}_{\mathrm{C}}\mathbf{R}$ respectively. It then computes the initial homography matrix $\mathbf{H}$ using the camera intrinsics $\mathbf{K}$ per (2) [44]:

$$\mathbf{H} = \mathbf{K}\,{}^{\mathrm{G}}_{\mathrm{C}}\mathbf{R}\mathbf{K}^{-1} \tag{2}$$

B. Horizon Cutoff

The homography procedure estimates the horizon cutoff row ρ in image space as illustrated in Fig. 4. It uses the successfully matched stereo points shown in Fig. 4b to find the first image row with a large enough ratio of valid matches. Rows above the horizon estimate are either empty space or distant landscape, unable to be correlated during stereo matching, which are not useful for future image feature matching in warped space.

If stereo matching struggles and no valid ρ is detected, LuVo uses the pitch angle of the camera with respect to the ground plane γ to estimate ρ per (3):

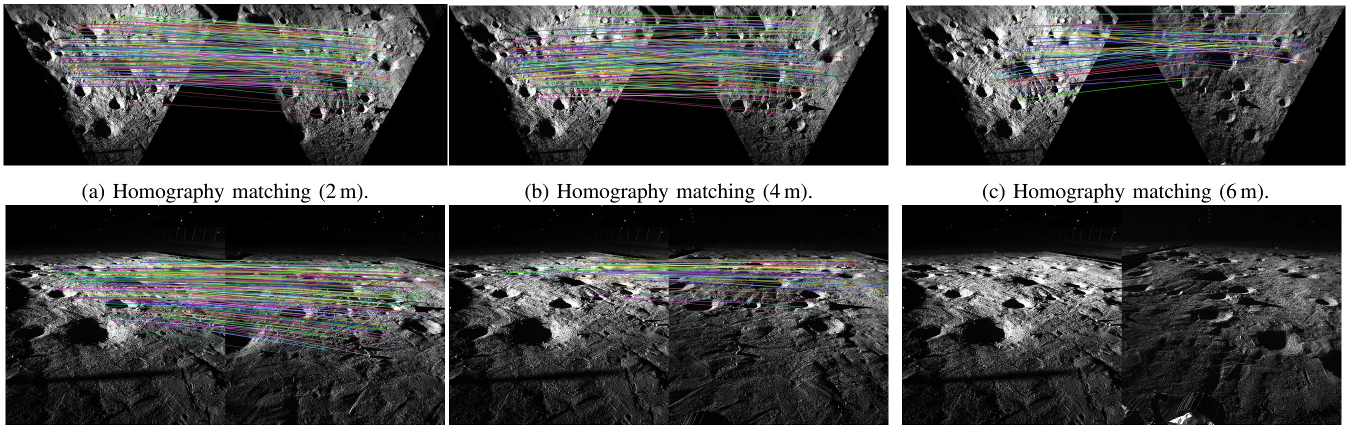
$$\rho = c_y - f_y \tan\gamma - \gamma_{\text{min}} \tag{3}$$

Here c_y and f_y are the y components of the camera principal point and focal length and γ_{min} is a minimum pitch angle threshold to ensure some amount of cutoff occurs.

C. Maximizing Projection Area

The procedure then creates a set of bounding coordinates $\vec{b}_i$ using the image corners defined by the horizon cutoff row and bottom of the image. To find the bounds of the projected image, it projects each corner into warped space per (4):

$$\vec{w}_i = \alpha(\mathbf{H}\zeta(\vec{b}_i)) \tag{4}$$



(a) Homography matching (2 m).

(b) Homography matching (4 m).

(c) Homography matching (6 m).

(d) Image space matching (2 m).

(e) Image space matching (4 m).

(f) Image space matching (6 m).

Fig. 5. Homography-based matching succeeds in finding correspondences at the NASA Roverscape facility [43] for stereo pairs separated by two, four, and six meters. Image space matching however gets fewer matches for the sequence with four meters of separation and fails to find matches at six meters.

where α applies homogeneous normalization and ζ converts a two dimensional vector to homogeneous coordinates.

As a last step, it calculates an offset and scale matrix to maximize the image coverage in warped space. The offset matrix $\mathbf{O}$ is calculated per (5):

$$\begin{bmatrix} 1 & 0 & -x_{\text{min}} \\ 0 & 1 & -y_{\text{min}} \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where x_{min} and y_{min} are the minimum x and y values of the projected coordinates $\vec{w}_i$. The adaptive homography matrix used for image warping is then calculated per (6):

$$\mathbf{H}_a = \mathbf{SOH} \tag{6}$$

Here $\mathbf{S}$ is the diagonal scale matrix. The x and y components are $\frac{r}{d}$, where r is the desired warped image resolution and d is the maximum of the x and y dimensions of the projected corners $\vec{w}_i$, while the z component is set to 1.

V. Image Feature Matching

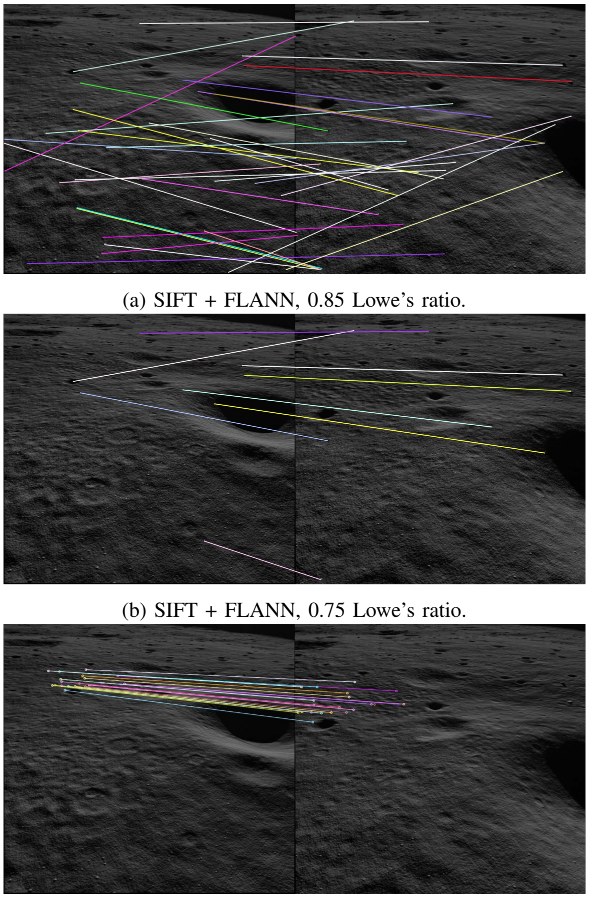
The LightGlue feature matcher in Fig. 3 uses LightGlue [10] with DISK image features [45] to find correspondences between images and follows this with an outlier rejection policy to further refine matches.

A. Classical versus Machine Learning Matching

Whereas classical image feature matching methods typically use nearest neighbor estimation [46] to find the closest descriptor match for each feature point individually, LightGlue finds matches as a set. As shown in Fig. 6, LightGlue greatly outperforms classical matching in repetitive environments like the lunar surface, where many false matches exist for nearest neighbor methods.

B. Outlier Rejection

The matcher removes outliers using MAGSAC++ [48] based fundamental matrix estimation. To increase the likelihood of detecting a sufficient number of inlier matches,



(a) SIFT + FLANN, 0.85 Lowe's ratio.

(b) SIFT + FLANN, 0.75 Lowe's ratio.

(c) LightGlue, 0.9 confidence.

Fig. 6. SIFT [47] matching with FLANN nearest neighbor estimation struggles on a pair of simulated lunar images separated by primarily horizontal movement, shown side by side in the above figures with detected matches drawn using colored lines. Using 2048 keypoints and Lowe's ratio test with a threshold of 0.85 results in more matches, but also more outliers as evident by the many crossed match lines in Fig. 6a . Using a lower ratio test threshold reduces the number of outlier matches as shown in Fig. 6b but misses many valid matches detected using LightGlue in Fig. 6c.

it inversely scales the model fitter inlier threshold based on the number of matches found by LightGlue. If fewer than 50 inlier matches are detected, all matches are discarded. It keeps up to 200 inlier matches, filtered using their LightGlue confidences.

VI. Point Cloud Alignment

The point cloud alignment procedure for LuVo, shown in Fig. 3, uses the set of image feature match pairs m_i computed by the feature matcher to align successive point clouds. For each image space point within a pair, it checks if a valid 3D point exists in the stereo cloud index corresponding to the point. If this fails, it queries points in a bounded search window around the feature point and chooses the closest match using an image space Manhattan distance metric, or discards the match pair if no valid points are found. It then computes the relative pose ${}^{A}_{B}\mathbf{T}$ between clouds by aligning the 3D point matches using the Umeyama algorithm [49]. The alignment procedure optionally further refines the relative pose using point-to-plane ICP initialized with ${}^{A}_{B}\mathbf{T}$.

VII. Results

We evaluate LuVo using a dataset of 155 odometry pairs containing stereo point clouds and images generated using the VIPER lunar simulator [50] for driving ranges displayed in Table I. Example images are shown in Figs. 1, 4 and 6. The dataset spans both expected driving distances under eight meters and further distances up to 16 meters. We use an Intel i9-12900HK 3.2 GHz CPU and NVIDIA RTX A2000 GPU for the evaluations.

TABLE I. Dataset Driving Distances

Translation Range [m]	0-2	2-4	4-6	6-8	8-10	10-12	12-14	14-16
Count	16	30	27	24	21	17	13	7

A. Methods

We compare the performance of LuVo to point-to-plane ICP using both groundtruth initialization (ICP Groundtruth Init.) and initialization with Gaussian noise of 10% for translations and 1% for rotations (ICP Noisy Init.). We use a larger value for translation noise to emulate wheel slip expected on the Lunar surface and a smaller rotation error due to the availability of star tracker orientation fixes for the rover.

Additionally, we compare to LuVo using image space matching (LuVo Img. Space) and LuVo followed by point-to-plane ICP (LuVo + ICP), where the estimated pose from LuVo is subsequently used as an initial pose for the ICP method.

We also provide an analysis of point-to-plane ICP performance for expected driving distances from 4-8 m using increasing noise profiles, again using 1% rotation error and increased translation noise from 0-35% in 5% increments.

For LightGlue matching we use nine layers, 2048 keypoints, a confidence threshold of 0.9, and DISK image features. For ICP we use the point-to-plane implementation from libpointmatcher [51] with an iteration limit of 60 and trimmed distance outlier filtering. We remove cloud points beyond 25 meters and add statistical outlier filtering, along with voxel and normal-based downsampling, to make normal computation and ICP correspondence estimation more tractable.

B. Evaluation Metrics

We measure both the absolute trajectory error (ATE) [52] and absolute rotation error (ARE) [53] for accuracy analysis, along with the success rate (SR) consisting of the percentage of successfully localized images within a defined threshold (0.3 m, 5°) [54]. We segment the evaluation based on the driving distances in Table I to analyze performance for different translation ranges.

C. ICP Accuracy vs. Noise

ICP accrues error and suffers a reduced success rate with increasing initialization noise as displayed in Fig. 7. ICP is able to accurately estimate poses when initialization noise is low, below 10% translation. Increasing error above this rate, however, begins to degrade pose estimation, and for especially noisy sequences success rates reduce to below 25%. For difficult lunar terrain where wheel slip may result in error above ∼15%, translation distances need to be limited to ensure ICP is properly initialized and capable of providing reliable estimates.

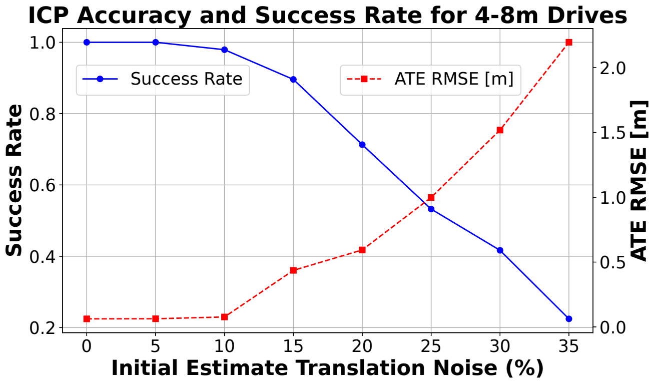


Fig. 7. RMSE position errors and success rates for ICP with different degrees of pose initialization noise using stereo pairs separated by 4-8 m.

D. Odometry Results

We show odometry accuracy results for the LuVo and ICP methods in Fig 8 and success rates in Fig. 9. Results are segmented based on each of the driving ranges depicted in Table I. Additionally, we summarize the results across all datasets in Table II. LuVo (Img. Space) accuracy in the summary only includes drives below 10 meters as it suffered failures for all driving distances above this.

1) Homography vs. Image Space Matching: LuVo using image space matching failed for drives above 10 meters and displayed an ATE 9.5× larger and a success rate 42% smaller than LuVo for drives below six meters. As shown in Fig. 8 and Fig. 9, LuVo maintains a success rate above 80% for

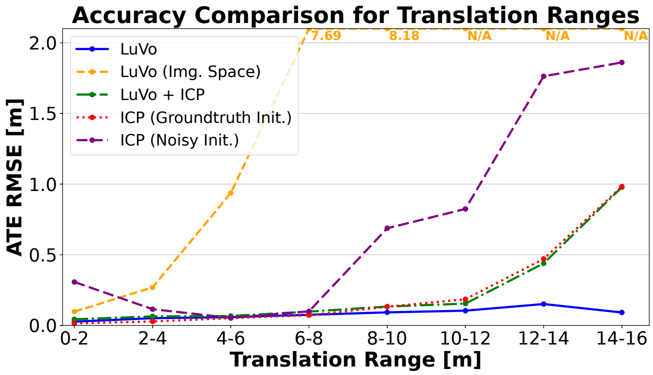


Fig. 8. RMSE position errors for LuVo and ICP methods categorized by translation range.

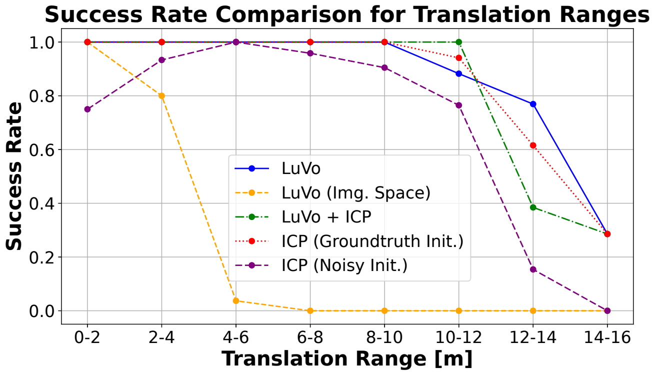


Fig. 9. Success rates for LuVo and ICP methods categorized by translation range.

drives up to 10 meters and outperforms image spaced matching for each driving range, demonstrating the performance improvements of using homography-based matching.

2) LuVo vs. ICP: LuVo outperforms other LuVo variants and ICP methods in ATE RMSE, ARE RMSE, and success rate on the lunar dataset as depicted by Table II. LuVo, LuVo + ICP, and ICP (Groundtruth Init.) perform comparably well for driving ranges below six meters. Above six meters, LuVo + ICP and ICP (Groundtruth Init.) begin to accrue more error while LuVo maintains an ATE RMSE below 0.2, even for driving distances beyond 14 meters. LuVo matching provides either high accuracy correspondences or too few correspondences for larger baseline images, making its successful pose estimates more reliable than ICP methods. The sparsity of stereo clouds at further distances, as illustrated in Fig. 1b, prevents ICP from outperforming LuVo, and even degrades LuVo's performance as evident by LuVo + ICP reducing accuracy and success rate for longer range drives compared to simply using LuVo.

ICP using 10% translation noise and 1% rotation noise demonstrates reduced success for drives below six meters compared to LuVo and increased ATE RMSE for drives above six meters. As discussed in §VII-C, initialization noise reduces ICP's accuracy whereas LuVo's initialization-free pose estimation enables it to more reliably perform VO in difficult driving terrain.

TABLE II. Odometry Comparison on Lunar Dataset

Method	ATE RMSE [m]	ARE RMSE [deg]	Avg. SR	Avg. Runtime [s]
LuVo	**0.081**	**0.378**	**0.867**	**0.61**
LuVo (Img. Space)	3.433*	1.819*	0.230	0.62
LuVo + ICP	0.247	1.136	0.834	2.17
ICP (Groundtruth Init.)	0.242	1.027	0.855	1.54
ICP (Noisy Init.)	0.714	3.362	0.683	1.61

3) Computation: LuVo runs ∼2.5 times faster than ICP-based approaches by leveraging the GPU for LightGlue matching. Unlike ICP, it does not require point cloud processing, including normal estimation, that takes ∼1.34 seconds per cloud.

VIII. Conclusion

We have presented LuVo, a lunar stereo visual odometry method for robust, long-distance relative pose estimation. By projecting navigation images onto the planar lunar surface and estimating a horizon cutoff to increase the usable matching area, LuVo is able to reliably find image feature matches between stereo clouds separated by several meters. It registers clouds using detected matches without relying on initial pose estimation, which is prone to wheel slip on the lunar surface.

We have demonstrated LuVo's improved performance on a dataset of simulated stereo sequences. While point-to-plane ICP results degrade as initialization noise and translations increase, LuVo steadily provides reliable and accurate poses for driving ranges up to 14 meters. It reduces error for drives above six meters even compared to ICP with perfect initialization, which loses information for matching due to increased sparsity in farther ranges of stereo clouds.

In future work, we wish to evaluate LuVo using Martian data, which shares many of the same localization challenges as the lunar surface. We would also like to incorporate the detection of non-planar obstacles and test performance for smaller baseline images using sustained feature tracks to increase LuVo's utility. We plan to use LuVo during the VIPER mission and anticipate reliable visual odometry performance for the rover's long-distance drives.

IX. Acknowledgements

We would like to thank the rest of the VIPER rover ground software, flight software, and simulation teams for supporting this work. This work was authored by employees of KBR Wyle Services, LLC under Contract No. 80ARC020D0010 with the National Aeronautics and Space Administration. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, or allow others to do so, for United States Government purposes. All other rights are reserved by the copyright owner.

REFERENCES

[1] A. Colaprete, "Volatiles investigating polar exploration rover (viper)," 2021.

[2] A. V. Nefian, X. Bouyssounouse, L. Edwards, T. Kim, E. Hand, J. Rhizor, M. Deans, G. Bebis, and T. Fong, "Planetary rover localization within orbital maps," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 1628–1632.

[3] M. Maimone, Y. Cheng, and L. Matthies, "Two years of visual odometry on the mars exploration rovers," *Journal of Field Robotics*, vol. 24, no. 3, pp. 169–186, 2007.

[4] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. Spie, 1992, pp. 586–606.

[5] K.-L. Low, "Linear least-squares optimization for point-to-plane icp surface registration," *Chapel Hill, University of North Carolina*, vol. 4, no. 10, pp. 1–3, 2004.

[6] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[7] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[8] J. Kruger, A. Rogg, and R. Gonzalez, "Estimating wheel slip of a planetary exploration rover via unsupervised machine learning," in *2019 IEEE Aerospace Conference*. IEEE, 2019, pp. 1–8.

[9] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4938–4947.

[10] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "Lightglue: Local feature matching at light speed," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 17 627–17 638.

[11] Y. Ma, S. Liu, B. Sima, B. Wen, S. Peng, and Y. Jia, "A precise visual localisation method for the chinese chang'e-4 yutu-2 rover," *The Photogrammetric Record*, vol. 35, no. 169, pp. 10–39, 2020.

[12] S. Mane, "Chandrayaan-2: India's lunar exploration mission to the moon," *International Journal of All Research Education and Scientific Methods*, vol. 11, no. 7, pp. 1116–1123, 2023.

[13] J.-P. de la Croix, F. Rossi, R. Brockers, D. Aguilar, K. Albee, E. Boroson, A. Cauligi, J. Delaune, R. Hewitt, D. Kogan, *et al.*, "Multi-agent autonomy for space exploration on the cadre lunar technology demonstration," in *2024 IEEE Aerospace Conference*. IEEE, 2024, pp. 1–14.

[14] J. M. Coughlan and A. L. Yuille, "Manhattan world: Compass direction from a single image by bayesian inference," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. IEEE, 1999, pp. 941–947.

[15] J. Wang, J. Li, S. Wang, T. Yu, Z. Rong, X. He, Y. You, Q. Zou, W. Wan, Y. Wang, *et al.*, "Computer vision in the teleoperation of the yutu-2 rover," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 3, pp. 595–602, 2020.

[16] M. Wagner, D. Wettergreen, and P. Iles, "Visual odometry for the lunar analogue rover "artemis"," in *ISAIRAS*, 2012.

[17] C. Tomasi and T. Kanade, "Detection and tracking of point," *Int J Comput Vis*, vol. 9, no. 137-154, p. 3, 1991.

[18] L. Li, J. Lian, L. Guo, and R. Wang, "Visual odometry for planetary exploration rovers in sandy terrains," *International Journal of Advanced Robotic Systems*, vol. 10, no. 5, p. 234, 2013.

[19] S. Daftry, Z. Chen, Y. Cheng, S. Tepsuporn, S. Khattak, L. Matthies, B. Coltin, U. Naal, L. M. Ma, and M. Deans, "Lunarnav: Crater-based localization for long-range autonomous lunar rover navigation," in *2023 IEEE Aerospace Conference*. IEEE, 2023, pp. 1–15.

[20] A. Cauligi, R. M. Swan, H. Ono, S. Daftry, J. Elliott, L. Matthies, and D. Atha, "Shadownav: Crater-based localization for nighttime and permanently shadowed region lunar navigation," in *2023 IEEE Aerospace Conference*. IEEE, 2023, pp. 1–12.

[21] D. T. Chelmins, B. W. Welch, O. S. Sands, and B. V. Nguyen, "A kalman approach to lunar surface navigation using radiometric and inertial measurements," Tech. Rep., 2009.

[22] V. Verma, M. W. Maimone, D. M. Gaines, R. Francis, T. A. Estlin, S. R. Kuhn, G. R. Rabideau, S. A. Chien, M. M. McHenry, E. J. Graser, *et al.*, "Autonomous robotics is driving perseverance rover's progress on mars," *Science Robotics*, vol. 8, no. 80, p. eadi3099, 2023.

[23] D. S. Bayard, D. T. Conway, R. Brockers, J. H. Delaune, L. H. Matthies, H. F. Grip, G. B. Merewether, T. L. Brown, and A. M. San Martin, "Vision-based navigation for the nasa mars helicopter," in *AIAA Scitech 2019 Forum*, 2019, p. 1411.

[24] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[25] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[26] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[27] D. Gálvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[28] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[29] S. Leutenegger, "Okvis2: Realtime scalable visual-inertial slam with loop closure," *arXiv preprint arXiv:2202.09199*, 2022.

[30] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2548–2555.

[31] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 15–22.

[32] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers, "Visual-inertial mapping with non-linear factor recovery," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 422–429, 2019.

[33] 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.

[34] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1689–1696.

[35] M. Abate, Y. Chang, N. Hughes, and L. Carlone, "Kimera2: Robust and accurate metric-semantic slam in the real world," *arXiv preprint arXiv:2401.06323*, 2024.

[36] L. Carlone, Z. Kira, C. Beall, V. Indelman, and F. Dellaert, "Eliminating conditionally independent sets in factor graphs: A unifying perspective based on smart factors," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 4290–4297.

[37] B. Guan, P. Vasseur, C. Demonceaux, and F. Fraundorfer, "Visual odometry using a homography formulation with decoupled rotation and translation estimation using minimal solutions," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2320–2327.

[38] O. Saurer, F. Fraundorfer, and M. Pollefeys, "Homography based visual odometry with known vertical direction and weak manhattan world assumption," in *Vicomor Workshop at IROS*, vol. 2012, 2012.

[39] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, *et al.*, "Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera," in *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 2011, pp. 559–568.

[40] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "Elasticfusion: Dense slam without a pose graph." in *Robotics: science and systems*, vol. 11. Rome, Italy, 2015, p. 3.

[41] J. Zhang, M. Kaess, and S. Singh, "Real-time depth enhanced monocular odometry," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 4973–4980.

[42] J.-Y. Bouguet *et al.*, "Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm," *Intel corporation*, vol. 5, no. 1-10, p. 4, 2001.

[43] M. Bualat, W. Carey, T. Fong, K. Nergaard, C. Provencher, A. Schiele, P. Schoonejans, and E. Smith, "Preparing for crew-control of surface robots from orbit," in *Space Exploration Conference*, no. ARC-E-DAA-TN12354, 2014.

[44] R. Szeliski, *Computer vision: algorithms and applications*. Springer Nature, 2022.

[45] M. Tyszkiewicz, P. Fua, and E. Trulls, "Disk: Learning local features with policy gradient," *Advances in Neural Information Processing Systems*, vol. 33, pp. 14 254–14 265, 2020.

[46] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration." *VISAPP (1)*, vol. 2, no. 331-340, p. 2, 2009.

[47] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, pp. 91–110, 2004.

[48] D. Barath, J. Noskova, M. Ivashechkin, and J. Matas, "Magsac++, a fast, reliable and accurate robust estimator," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1304–1312.

[49] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 13, no. 04, pp. 376–380, 1991.

[50] M. Allan, U. Wong, P. M. Furlong, A. Rogg, S. McMichael, T. Welsh, I. Chen, S. Peters, B. Gerkey, M. Quigley, *et al.*, "Planetary rover simulation for lunar exploration missions," in *2019 IEEE Aerospace Conference*. IEEE, 2019, pp. 1–19.

[51] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP Variants on Real-World Data Sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, Feb. 2013.

[52] J. Sturm, N. Engelhard, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proc. of IROS*, 2012.

[53] P. Kim, B. Coltin, O. Alexandrov, and H. J. Kim, "Robust visual localization in changing lighting conditions," in *IEEE ICRA*, 2017.

[54] W. Wang, D. Zhu, X. Wang, Y. Hu, Y. Qiu, C. Wang, Y. Hu, A. Kapoor, and S. Scherer, "Tartanair: A dataset to push the limits of visual slam," in *IEEE IROS*, 2020.