High-fidelity Synthetic Data Generation for Lunar Exploration via Physically-based Simulation

Ludovica Cavalieri
Department of Mechanical and Aerospace Engineering
Sapienza, University of Rome
Rome, Italy
ludovica.cavalieri@uniroma1.it

Mohamed El Awag
Department of Mechanical and Aerospace Engineering
Sapienza, University of Rome
Rome, Italy
mohamed.elawag@uniroma1.it

Simone Andolfo
Department of Mechanical and Aerospace Engineering
Sapienza, University of Rome
Rome, Italy
simone.andolfo@uniroma1.it

Fabio Valerio Buonomo
Department of Mechanical and Aerospace Engineering
Sapienza, University of Rome
Rome, Italy
fabiovalerio.buonomo@uniroma1.it

Riccardo Teodori
Department of Mechanical and Aerospace Engineering
Sapienza, University of Rome
Rome, Italy
riccardo.teodori@uniroma1.it

Antonio Genova
Department of Mechanical and Aerospace Engineering
Sapienza, University of Rome
Rome, Italy
antonio.genova@uniroma1.it

***Abstract*—As lunar exploration missions gain renewed interest, driving the need for advanced autonomous systems, the limitations of current methods become evident. The integration of artificial intelligence in the navigation systems of space probes has proven to be a reliable solution to improve the efficiency and robustness of guidance, navigation, and control systems. However, these approaches rely on large, high-quality training datasets, which cannot be easily acquired in space environments. Synthetic datasets have emerged as a viable alternative to mitigate this issue, enabled by the recent advances in photorealistic simulation tools. Nevertheless, models trained solely on synthetic data generally fail to generalize well to real imagery. To mitigate this effect, we developed a unified framework combining physical realism, rendering fidelity, and flexible annotation across multiple mission scenarios using Blender, a 3D modeling suite that supports physically grounded photorealistic rendering. The framework is evaluated on representative space robotics tasks, such as terrain classification and slope computation. The results demonstrate that the simulator can serve as a dataset generator and testbed for advanced perception pipelines, supporting the development of AI-enhanced navigation pipelines for different space exploration scenarios.**

***Index Terms*—space robotics and automation, data sets for robotic vision, deep learning for visual perception**

I. Introduction

Lunar exploration missions have gained renewed momentum, driven by governmental initiatives such as NASA's Artemis program and the CNSA's Chang'e missions, alongside a growing number of commercial efforts. These missions require advanced autonomous systems capable of operating in the harsh, unstructured lunar environment, where perception is a key enabler for safe navigation, landing, and surface operations.

Recent advances in Artificial Intelligence (AI) have demonstrated the potential of integrating Machine Learning (ML) algorithms into the navigation systems of robotic platforms for space exploration. A major drawback of these approaches is their reliance on large, high-quality annotated datasets, which are challenging to acquire in the space domain due to the limited availability of *in situ* imagery and the time-intensive nature of manual labeling. Synthetic datasets have emerged as a viable alternative, enabled by the ever-increasing capabilities of photorealistic simulation tools. However, models trained purely on synthetic data often exhibit performance discrepancies when applied to real images. Bridging the *sim-to-real* gap requires integrating photorealistic rendering, automated annotation, and physically based environment modeling to maximize visual and geometrical similarity between simulation and real world.

Numerous simulators have been developed for planetary environments, but most focus either on physically-consistent surface modeling or on automated data annotation. End-to-end tools that combine both elements remain limited and are often designed for specific applications. Among the most scientifically rigorous simulators are those based on the Planet and Asteroid Natural Scene Generation Utility (PANGU) [1], widely used in ESA projects for planetary scene modeling and multi-sensor simulation. Conversely, simulators such as OAISYS (Outdoor Artificial Intelligent SYstems Simulator) [3] combine photorealistic rendering with

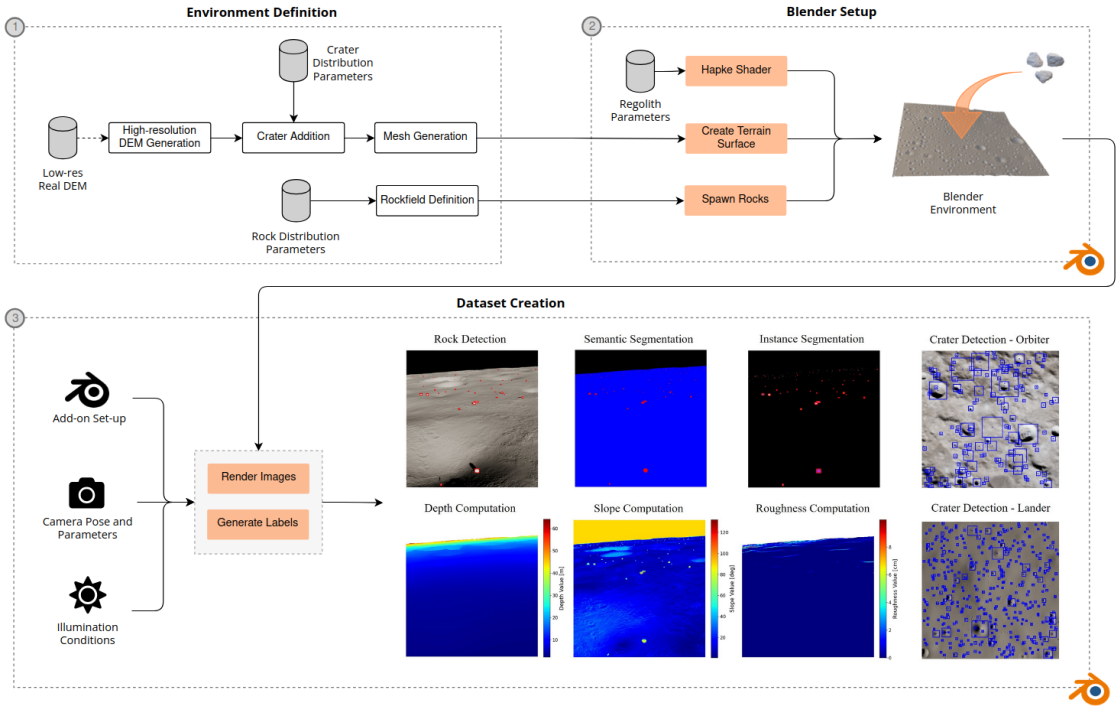

Fig. 1. Schematic overview of the proposed simulator.

automated annotation and are effective for large-scale dataset generation. However, their scene construction is not grounded in geological processes, limiting their ability to reproduce realistic planetary surface morphology and to mitigate the sim-to-real gap. CORTO (Celestial Object Rendering TOol) [4] combines realistic environment generation with multi-modal annotation capabilities, but is primarily designed for small-body proximity operations and lunar orbital scenarios, limiting its use for surface exploration and landing.

In this work, we present a novel Blender-based simulation framework for lunar environments supporting surface, landing, and orbital mission scenarios. The simulator combines physically grounded terrain and regolith modeling, photorealistic rendering, and automated annotation, allowing the generation of large-scale, high-fidelity synthetic datasets for a wide range of perception tasks, including semantic and instance segmentation, object detection, depth estimation, and slope and roughness computation.

II. Simulation Framework

The Blender-based framework, built on our previous work [5], enables the generation of high-fidelity datastes for lunar exploration. Thanks to its physics-based rendering engine, *Cycles*, Blender [6] supports the production of high-quality optical data, further enhanced through the implementation of custom materials using the Open Shading Language (OSL) [7].

Figure 1 shows an overview of the software architecture. The framework integrates elevation data from past lunar missions with fractal augmentation algorithms and crater and rock size-frequency distributions to produce high-fidelity environments, ensuring visual and geometric realism. These environments are imported into Blender and assigned physically-consistent materials. An OSL-based shader is used to simulate the reflectivity of lunar regolith, while Cycles and a custom add-on built on Blender's rendering passes system are used to generate high-quality optical data and user-selected annotations for downstream AI annotations. Compared to [5], which focused on rover surface operations, the framework has been extended to support lander- and satellite-based observation geometries [8]. Visual fidelity is further enhanced through the use of albedo maps in the regolith material implementation [9]. Finally, the add-on was integrated with the capability for crater annotation in orbiter and lander scenarios.

A. Planetary Surface Modeling

The first step of the simulation pipeline involves generating a 3D representation of the lunar environment from existing elevation data. Due to the computational cost of global mesh generation, the framework only converts into a mesh a patch of the Digital Elevation Model (DEM) of user-defined size. For landing and orbital scenarios, the patch size is derived from probe altitude and camera Field-of-View (FOV), ensuring full scene coverage. This is achieved by computing the intersection between the viewing rays corresponding to the image boundaries and the Moon, modeled as a sphere.

Although DEMs exist for the entire lunar surface, their resolution is sufficient only for simulating high-altitude observations. In contrast, rovers and landing platforms operate in close proximity to the surface, making high-resolution

elevation data essential for precise hazard simulation. To address this limitation, we developed a dedicated pipeline for synthesizing high-resolution lunar terrains. The terrain augmentation pipeline is based on the *diamond-square* algorithm, a Random Mid-Point Displacement (RMD) method that computes new elevation values by interpolating existing data in alternate diamond and square phases [10]. To mitigate the artifacts associated with this approach, we decomposed the terrain in low- and high-frequency components, and applied the diamond-square algorithm to the high-frequency component only, while upsampling the low-frequency component using a bilinear interpolation scheme, thereby significantly diminishing the severity of the artifacts [11].

However, resolution augmentation alone is insufficient to reproduce the appearance of the lunar surface at close range, as it does not introduce new information and cannot recover small-scale features that cannot be detected in the low-resolution data acquired from orbit, which are critical for accurately simulating hazards in rover and landing scenarios. To address this, crater and rock densities are modeled through established cumulative size-frequency distributions [12]. Additionally, crater profiles are computed using a piecewise polynomial function to account for degradation and age [13], and rock meshes are sourced from NASA's Astromaterials 3D dataset [14] and placed at desired locations within Blender.

B. Lunar Regolith Simulation

Standard models cannot replicate with sufficient fidelity the complex multiple-scattering effects induced by the regolith covering the lunar surface. Among the several Bidirectional Reflectance Distribution Functions (BRDFs) proposed to model regolith reflectance, the Hapke model emerges as the most complete and physically grounded, recognizing the discrete nature of regolith particles and incorporating parameters related to their roughness, density, and porosity [15]. Additionally, it captures the dependence of the regolith's appearance on the phase angle (*i.e.*, the angle between the observer and the solar rays), reproducing the characteristic bright appearance of the regolith at low phase angles and its darker shading at higher ones. Finally, it accounts for both single and multiple scattering within the medium, enabling the simulation of the opposition surge, a sharp increase in reflectance at small phase angles caused by coherent backscattering and shadow-hiding among particles.

By leveraging Blender's compatibility with OSL, it was possible to implement a custom shader based on the Hapke model, allowing accurate reproduction of the key photometric properties of the lunar surface, as shown in Figure 2. Surface texture is further refined using displacement and roughness textures sourced from the Poly Haven team [16]. In addition to the previous software version, albedo maps adapted from the Hapke-normalized Wide Angle Camera (WAC) mosaic are employed to achieve the typical lunar coloring and recreate its variations depending on geological composition, ensuring enhanced visual fidelity, especially at higher altitudes.

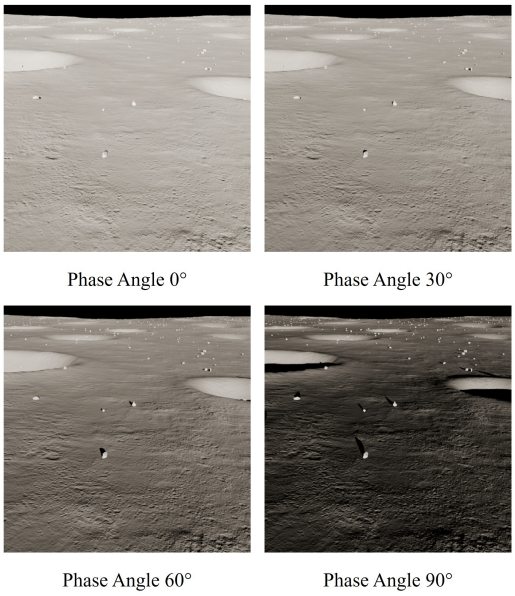

Fig. 2. Simulated lunar surface observed at different phase angles.

Representative rendering outputs generated using this setup are shown in Figure 3. From left to right, the Figure shows an image acquired from a nadir-pointing orbiting probe at an altitude of $200\ km$, a render acquired from a lander at an altitude of $500\ m$, and finally an image from a surface probe.

C. Automated Dataset Generation

Once the environments are generated and the observation geometry is defined, synthetic images are rendered using Blender Cycles, a path-tracing approach that models light as rays and maintains physical consistency in simulating its interactions with objects and the camera. Then, a custom add-on leveraging Blender's rendering passes system, and the knowledge of the simulated environment and observation geometry is used to generate multiple annotations for typical AI tasks.

Semantic segmentation masks are created by assigning the same float color to all objects belonging to the same class, while instance masks are obtained by assigning a unique color identifier to each object. Bounding boxes are then derived by clustering instance masks. These annotations are particularly important in surface driving scenarios, where knowledge of terrain types and rock location is crucial to support safe path planning operations.

Depth masks are achieved by using Blender's BVH acceleration structure to compute the distance at which the viewing rays intersect the scene. A similar ray-tracing pass is used to reconstruct the elevation profile within the camera FOV. The spatial gradients of the elevation profile, computed through Sobel filters, enable the definition of the slope map, while the second-order derivative, computed with a

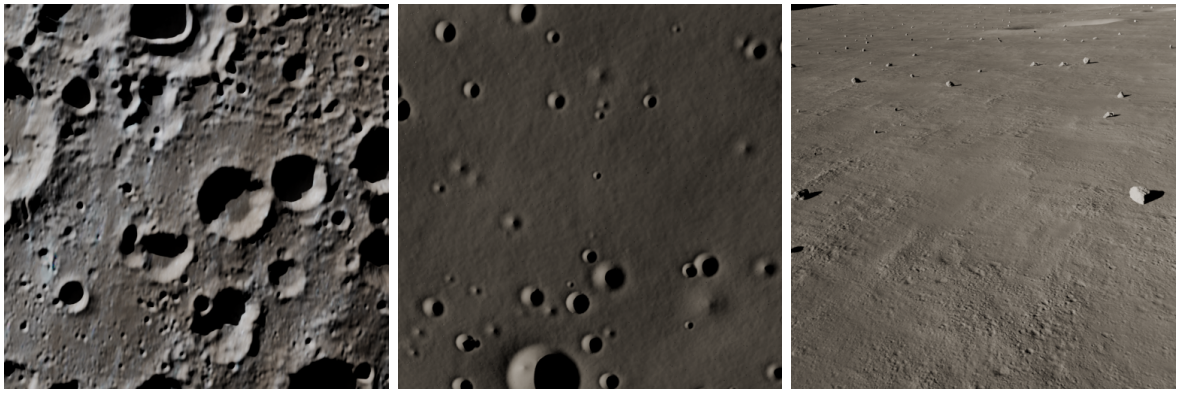

Fig. 3. Synthetic images generated for different mission scenarios. Left to right: orbiter - altitude 200 km, lander - altitude 500 m, rover.

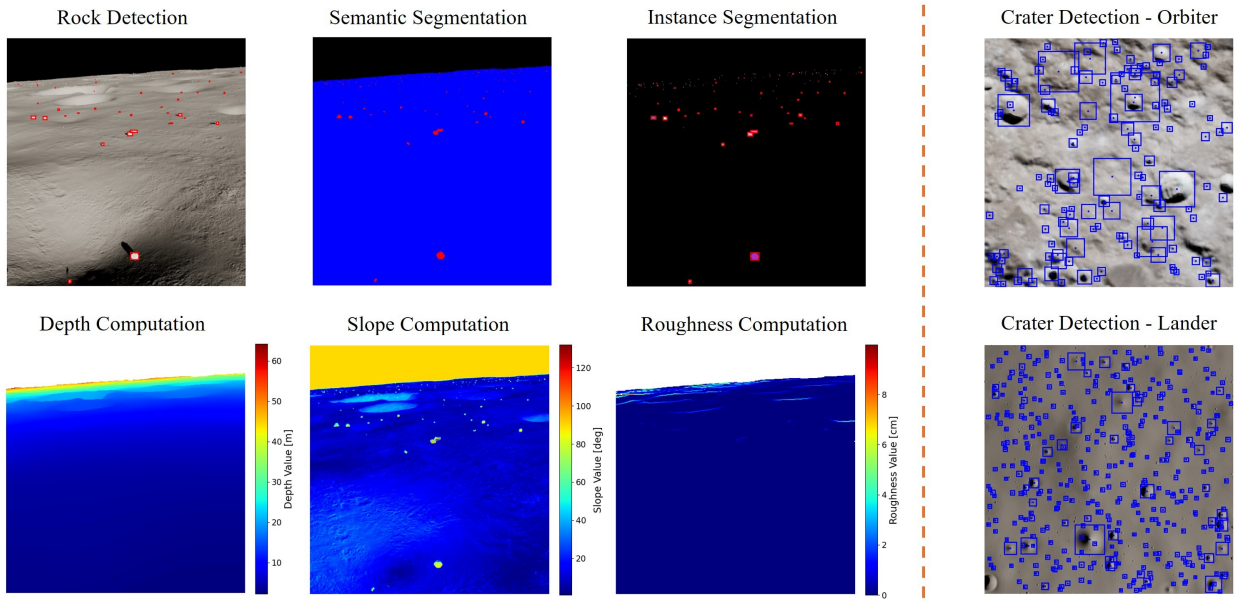

Fig. 4. Full set of annotations that can be generated with the simulator.

Laplacian filter, provides the roughness map. Landing and surface exploration operations can benefit from neural systems capable of efficiently characterizing the geometry of the observed scene. Finally, crater detection labels for orbital and landing missions were introduced. These labels are obtained by reprojecting in the image plane the coordinates of craters from the Robbins global database of impact craters [17] and from the small crater database generated during the crater addition phase in the terrain augmentation pipeline.

Figure 4 shows the full set of currently implemented labels. For clarity, most labels were produced for an example image acquired by a surface probe. However, crater detection labels (on the right) are shown for landing and orbital mission scenarios.

III. Simulator Testing

To assess the capability of the proposed framework to support the development and validation of advanced AI-aided perception pipelines, we selected a set of typical planetary robotics tasks. Namely, we evaluated the simulator's ability to support the training of segmentation modules for hazard detection in surface exploration scenarios, and we assessed its ability to aid the development of landing hazard detection and avoidance modules by training neural models to compute slope and roughness. Further validation can be found in [5], [18].

A. Semantic Segmentation for Surface Operations

Semantic segmentation can support the computation of safe paths to goal locations and the on-board selection of scientifically meaningful target sites by providing meaningful pixel-wise information on the explored environment. For this reason, we synthesized a dataset of 1000 images, 80% of which are used for training, and the rest are equally split between validation and testing. In a lunar environment, a minimal categorization is sufficient to support efficient and

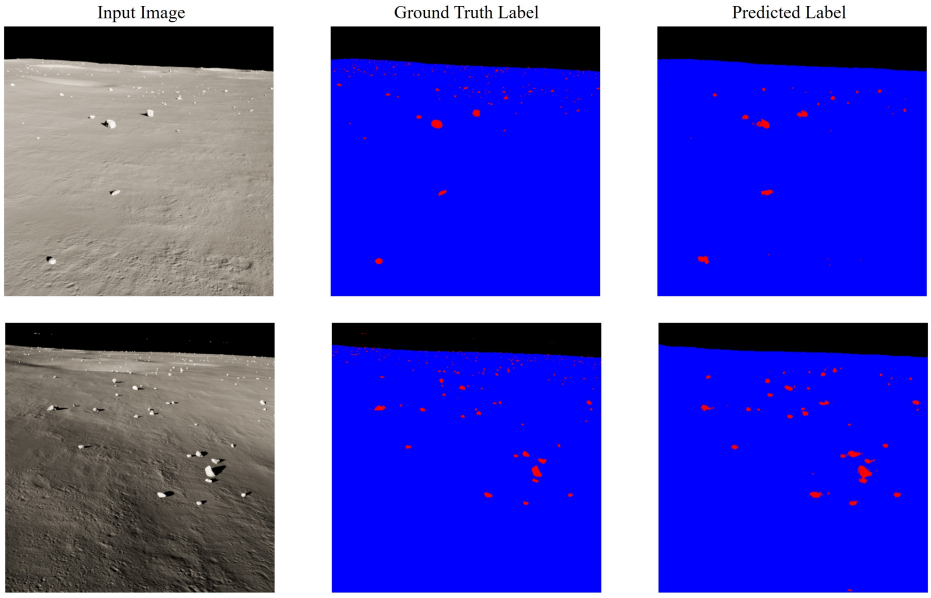

Fig. 5. UNet semantic segmentation results on synthetic renders.

informed path planning. For this reason, our dataset only contains three classes: sky, terrain, and rock. Using the Adam algorithm [19], we trained a state-of-the-art UNet [20] model with SegFormer backbone. Network performance was assessed using standard metrics, namely per-pixel accuracy, Intersection-over-Union, precision, and recall.

As shown in Figure 5 and Table I, the network performs well, producing annotations that are a close pixel-wise match to the ground truth labels, as reflected by the measured accuracy. Occasional misclassifications of the least represented class, *i.e.*, rock, may occur, lowering the network's IoU, precision, and recall scores. However, given that the dataset is inherently affected by a strong class imbalance, this behavior is to be expected.

TABLE I
SEMANTIC SEGMENTATION PERFORMANCE METRICS ON THE TEST SET.

Accuracy	IoU	Precision	Recall
0.9802	0.6850	0.8129	0.7210

B. Surface Slope Estimation for Landing Operations

Dense regression modules can be used to characterize the morphological properties of the terrain, including surface slope and roughness. This allows fine-grained characterization of the environment that can be used for safety analysis and to support landing site selection [18].

Within this context, we trained a UNet with Resnet50 backbone to perform slope estimation on lander-acquired imagery using the Adam algorithm. The performance of the model was evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Structural Similarity Index Measure (SSIM), and gradient MAE. This set of metrics allows a thorough description of the network's capability to match the target labels, and understand and replicate their structure and patterns.

As shown in Figure 6 and Table II, the network can largely replicate the structural patterns and the fine-grained details seen in the ground truth labels. Inference errors are present but confined to areas where prediction is inherently more challenging, such as crater rims, where variability is significant. Overall, the errors are significantly smaller than the inferred values, and thus do not compromise output interpretability for correct safety assessment.

TABLE II
DENSE REGRESSION PERFORMANCE METRICS ON THE TEST SET.

MAE [deg]	RMSE [deg]	SSIM	Gradient MAE [deg/pix]
4.1573	5.0564	0.7190	5.9409

IV. CONCLUSIONS

This work presents an end-to-end framework for high-fidelity synthetic dataset generation tailored to lunar scenarios for rover, lander, and orbiter missions. The proposed framework integrates scientifically grounded terrain modeling, physically based regolith simulation, photorealistic rendering, and automated data annotation. The software provides the visual and geometric realism needed to support the development and validation of advanced AI-aided perception modules for terrain assessment, hazard detection, target site identification, localization, and mapping.

The simulation tool was tested on representative tasks in surface operations and landing scenarios. The results constitute a preliminary assessment of the framework's capabilities and indicate that the simulator can be effectively employed to train and test ML algorithms to integrate in guidance, navigation, and control systems. Due to the scarcity of

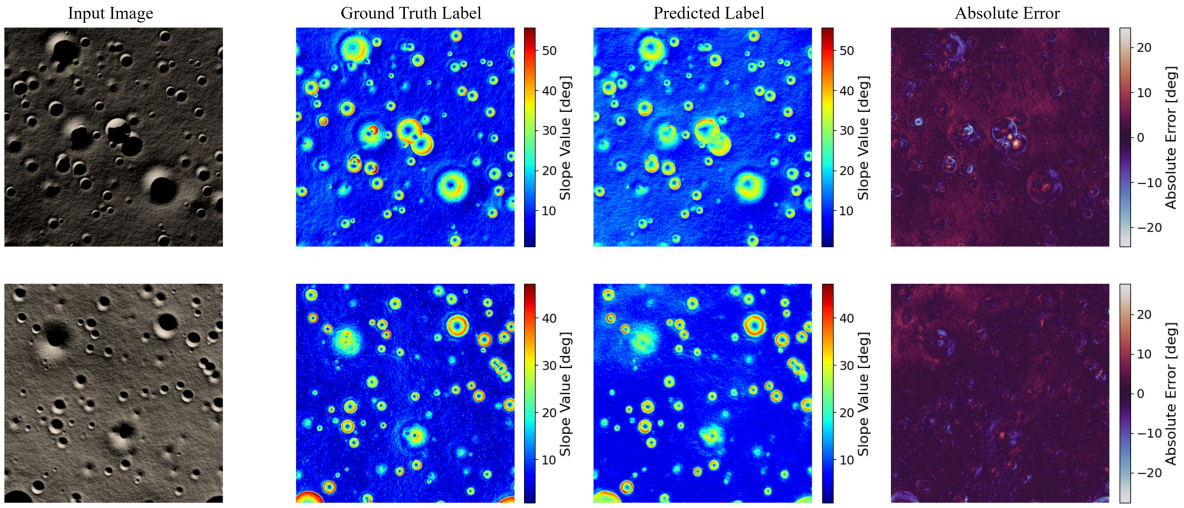

Fig. 6. UNet slope estimation results on synthetic renders.

real-world annotated datasets, its potential to bridge the sim-to-real gap remains to be fully assessed. Future research will focus on the rigorous evaluation of the images realism and the generalization capability of models trained on our synthetic datasets to real lunar imagery. This will provide critical information on how to extend and improve the simulation software to support the design and validation of deployment-ready algorithms that will perform well and reliably even in real-world scenarios.

Overall, the proposed framework represents a step toward the systematic generation of physically consistent, richly annotated datasets for planetary exploration, contributing to the advancement of robust and reliable AI-enhanced autonomous systems for future lunar missions.

REFERENCES

[1] S. Parkes, I. Martin, M. Dunstan, and D. Matthews, "Planet surface simulation with PANGU," in Proc. AIAA Conf., 2004. doi: 10.2514/6.2004-592-389.

[2] M. Allan, U. Wong, P. M. Furlong, A. Rogg, S. McMichael, T. Welsh, I. Chen, S. Peters, B. Gerkey, M. Quigley, M. Shirley, M. Deans, H. Cannon, and T. Fong, "Planetary rover simulation for lunar exploration missions," in Proc. IEEE Aerospace Conf., 2019, pp. 1–19. doi: 10.1109/AERO.2019.8741780.

[3] M. G. Müller, M. Durner, A. Gawel, W. Stürzl, R. Triebel, and R. Siegwart, "A photorealistic terrain simulation pipeline for unstructured outdoor environments," in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), 2021.

[4] M. Pugliatti, C. Buonagura, and F. Topputo, "CORTO: the celestial object rendering tool at DART lab," Sensors, vol. 23, no. 23, 2023. doi: 10.3390/s23239595.

[5] L. Cavalieri, F. V. Buonomo, S. Andolfo, M. El Awag, R. Teodori, and A. Genova, "A Physically Accurate Simulation Framework for Training and Validation of AI-Based Algorithms for Space Exploration," in Proc. AIAA SciTech Forum, 2026, Paper 2026-2041. doi: 10.2514/6.2026-2041.

[6] Blender Online Community, *Blender: a 3D modelling and rendering package*. Amsterdam, The Netherlands: Blender Foundation, 2018. [Online]. Available: http://www.blender.org

[7] L. Gritz, C. Stein, C. Kulla, and A. Conty, "Open shading language," in ACM SIGGRAPH 2010 Talks, New York, NY, USA: Association for Computing Machinery, 2010. doi: 10.1145/1837026.1837070.

[8] S. Andolfo, A. Genova, F. V. Buonomo, A. M. Gargiulo, M. El Awag, P. Federici, R. Teodori, R. La Grassa, C. Re, and G. Cremonese, "Neural network-aided optical navigation for precise lunar descent operations," *Aerospace*, vol. 12, no. 3, p. 195, 2025. doi: 10.3390/aerospace12030195.

[9] S. Andolfo, A. Genova, M. E. Awag, F. V. Buonomo, P. Federici, and R. Teodori, "A crater-based optical navigation approach for precise spacecraft localization," in Proc. IEEE Aerospace Conf., Big Sky, MT, USA, 2025, pp. 1–13. doi: 10.1109/AERO63441.2025.11068689.

[10] G. S. P. Miller, "The definition and rendering of terrain maps," in Proc. 13th Annu. Conf. Comput. Graph. Interact. Tech. (SIGGRAPH), New York, NY, USA: Association for Computing Machinery, 1986, pp. 39–48. doi: 10.1145/15922.15890.

[11] U. J. Shankar, W.-J. Shyong, T. B. Criss, and D. Adams, "Lunar terrain surface modeling for the ALHAT program," in Proc. IEEE Aerospace Conf., 2008, pp. 1–10. doi: 10.1109/AERO.2008.4526300.

[12] H. J. Melosh, *Impact cratering: a geologic process*. New York, NY, USA: Oxford University Press, 1989.

[13] I. Martin, S. Parkes, and M. Dunstan, "Modeling cratered surfaces with real and synthetic terrain for testing planetary landers," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2916–2928, 2014. doi: 10.1109/TAES.2014.120282.

[14] E. H. Blumenfeld *et al.*, "Research-grade 3D virtual astromaterials samples: novel visualization of NASA's Apollo lunar samples and Antarctic meteorite samples to benefit curation, research, and education," 2017.

[15] B. Hapke, *Theory of reflectance and emittance spectroscopy*. Cambridge, UK: Cambridge University Press, 2012.

[16] Poly Haven, "Scanning the moon," 2025. [Online]. Available: https://blog.polyhaven.com/moon/

[17] S. J. Robbins, "A new global database of lunar impact craters ¿1–2 km: 1. Crater locations and sizes, comparisons with published databases, and global analysis," *J. Geophys. Res.: Planets*, vol. 124, no. 4, pp. 871–892, 2019. doi: 10.1029/2018JE005592.

[18] M. El Awag, L. Cavalieri, S. Andolfo, F. V. Buonomo, and A. Genova, "AI-enhanced vision-based hazard detection operations in lunar landing scenario," in Proc. IEEE Aerospace Conf., Big Sky, MT, USA, Mar. 2026.

[19] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," in *Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham, Switzerland: Springer, 2015, pp. 234–241.