A Hybrid AI-assisted Hazard Detection and Avoidance Pipeline for Autonomous Lunar Landing

***Index Terms*—Space Robotics and Automation, AI-Enabled Robotics, Motion and Path Planning**

I. Introduction

Safe autonomous landing is a key enabling capability for the next generation of lunar exploration missions. These initiatives will target landing sites with increasing scientific interest but inherently hazardous, with regions characterized by extreme illumination conditions and highly irregular terrain [1]. These conditions pose significant challenges for autonomous descent and landing operations. Real-time environmental awareness is therefore required to replan the final descent trajectory based on the observed terrain properties, including potential hazards such as boulders, craters, and steep slopes [2].

Recent lunar landing attempts, such as those conducted within NASA's Commercial Lunar Payload Services program, including missions by Firefly Aerospace and Intuitive Machines [3], have highlighted both the progress made and the remaining challenges of such demanding operations. The development of autonomous onboard systems appears therefore pivotal to achieve reliable, high-precision autonomous landings.

In this context, conventional Guidance, Navigation, and Control (GNC) architectures, based on deterministic modeling, have demonstrated limitations in terms of robustness, adaptability, and real-time responsiveness [2]. As of today, their ability to cope with unstructured environments and late-stage hazard identification remains insufficient to guarantee safe and high-precision payload delivery. To address these challenges, advanced perception systems are required to extract and use information in real-time, enabling prompt hazard detection and trajectory replanning.

Despite significant progress, Hazard Detection and Avoidance (HDA) approaches still exhibit limitations in terms of computational efficiency and onboard deployability [4]. In particular, conventional methods struggle to process high-resolution terrain data, constraining the minimum detectable size of hazardous features [5]. Furthermore, the time-critical nature of descent imposes strict constraints on trajectory replanning, often leading to simplified geometry-based strategies that neglect optimality and mission constraints such as propellant efficiency and dynamic feasibility [6].

These limitations motivate the adoption of advanced methodologies capable of combining real-time performance with multi-objective optimization. In this context, recent advances in Artificial Intelligence (AI) offer a promising research direction to enhance onboard autonomy, enabling more efficient sensor data processing and near-optimal control synthesis, while maintaining the reliability levels required for safety-critical space applications.

II. Proposed Framework

This work proposes a hybrid AI-assisted pipeline for autonomous lunar landing, shown in Figure 1, integrating data-driven models with deterministic control techniques to address both Hazard Detection (HD) and Hazard Avoidance (HA). The approach shifts computationally intensive operations, otherwise infeasible for onboard execution, to learning-based models, where complexity is handled offline during training, enabling real-time execution during the terminal descent phase.

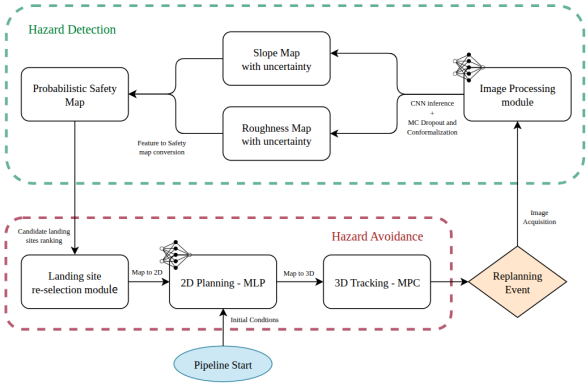


Fig. 1. Schematic representation of the presented HDA Pipeline. The modules integrating AI elements are highlighted with a network icon.

The HDA problem is inherently twofold. Hazard detection requires rapid processing of high-resolution terrain data to identify small-scale obstacles during terminal descent, imposing strict computational constraints on onboard processors. Hazard avoidance, instead, involves real-time trajectory replanning towards a safe landing site while accounting for mission-critical objectives such as propellant minimization and dynamic feasibility.

To address these challenges, the proposed framework combines Convolutional Neural Networks for vision-based terrain analysis, a Multi-Layer Perceptron (MLP) for rapid trajectory re-planning, and a Model Predictive Control (MPC) for robust trajectory tracking.

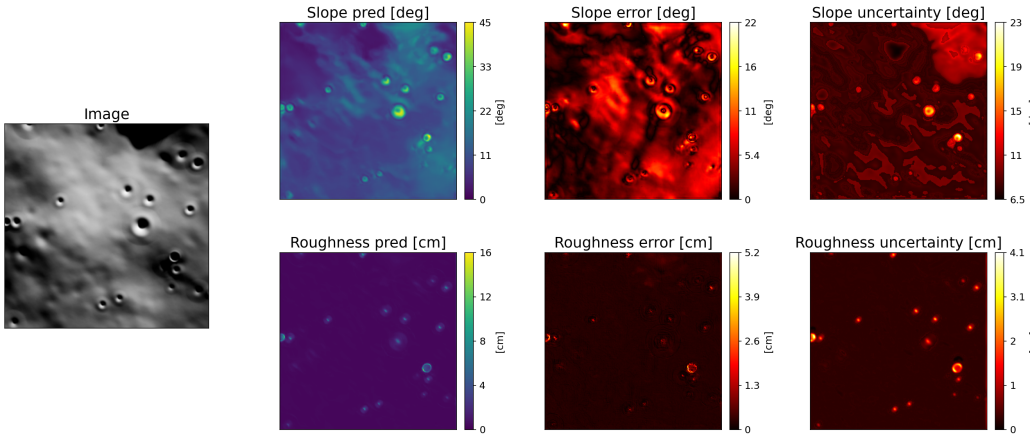


Fig. 2. Preliminary results of networks prediction for slope and roughness with associated errors and uncertainties. The input image is taken from 500 m of altitude by a nadir-pointing camera. Although conservative, the uncertainty provides a consistent representation of networks error.

III. Hazard Detection

In the HD phase, monocular grayscale images acquired during descent are processed using two separate U-Nets [7] to estimate continuous terrain features, namely slope and surface roughness [8]. This approach addresses the limitations of classification-based methods, which rely on discretized hazard levels, by providing spatially continuous high-resolution feature maps that enable more precise characterization of the landing area [9]. Predicted maps are paired with pixel-wise uncertainty estimates to construct a probabilistic landing safety map, enabling risk-aware evaluation of candidate landing sites [10].

The networks are trained on a synthetic dataset generated using a Blender-based photorealistic pipeline [11], where high-resolution terrain models are computed by using Digital Elevation Models derived by existing orbital data and enhanced through fractal refinement [12]. Ground-truth slope and roughness labels are computed via spatial filtering (e.g., Sobel and Laplacian operators). This fully automated process enables the generation of large-scale annotated datasets, which are otherwise unavailable from real lunar images.

Uncertainty estimation is addressed through a two-step process. Monte Carlo Dropout is used to estimate heuristic epistemic uncertainty first, via multiple stochastic forward passes with active dropout layers at inference time [13]. Conformal prediction is then applied to calibrate this uncertainty estimate [14]. A calibration dataset, similar to the testing one, is used to compute nonconformity scores, whose empirical quantile defines a correction factor used to adjust the predicted uncertainty bounds. This results in conservative coverage, distribution-free uncertainty estimates pivotal for safety-critical decision-making. Example of networks output, errors and conformalized uncertainties are shown in Figure 2.

Following the construction of the uncertainty-aware landing safety map, candidate landing sites are selected using a search and ranking procedure based on the Vehicle Dispersion Footprint Ellipse, which accounts for the physical dimensions of the lander [15]. Regions not satisfying a minimum safety threshold of 50% are discarded. The remaining candidates are ranked using a composite score that combines: (i) safety probability, (ii) maneuver cost, defined as the Euclidean distance from the current position, and (iii) a clearance score measuring distance from hazardous regions. The clearance score is computed using a Euclidean distance transform applied to a binary hazard mask, providing, for each pixel, the distance to the nearest hazardous area. An example of the safety map generation and landing site re-selection procedure is shown in Figure 3.

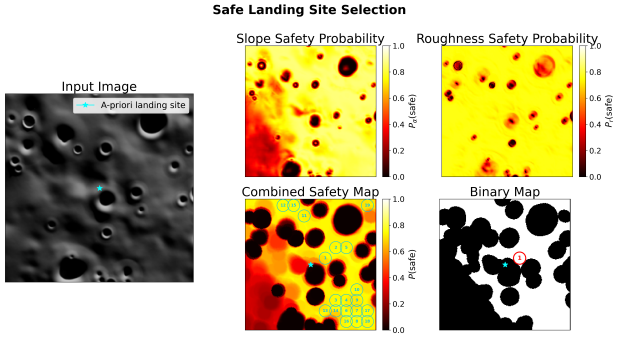


Fig. 3. Safety map generation and landing site re-selection for a landing image acquired from 500 m of altitude.

IV. Hazard Avoidance

Once a safe landing site has been selected by processing a given query image, the HA module computes a trajectory connecting the current state to the target. Traditional approaches typically rely on geometric guidance strategies, such as the Apollo polynomial guidance [16], which enable closed-form solutions and fast onboard evaluation. While computationally efficient, these methods are inherently limited in their ability to incorporate complex state and control constraints, and they generally do not allow for the explicit optimization of mission-relevant objectives, such as propellant

consumption. Conversely, numerical optimal control solvers are computationally prohibitive for onboard use. To overcome this, a Multi-Layer Perceptron (MLP) is trained offline to approximate the solution of a mass-optimal control problem derived from Pontryagin's Minimum Principle. Similarly to [17], the training dataset is generated using indirect optimization methods combined with homotopy techniques to guarantee convergence. Approximately 10000 fuel-optimal soft-landing trajectories were computed. The trajectories are later sampled and converted into thousands of state-action pairs as training samples (see Figure 4).

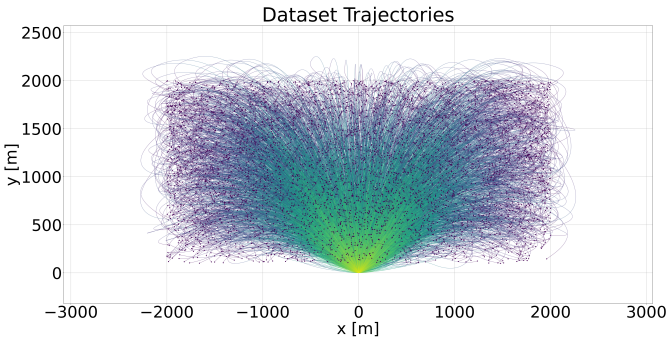
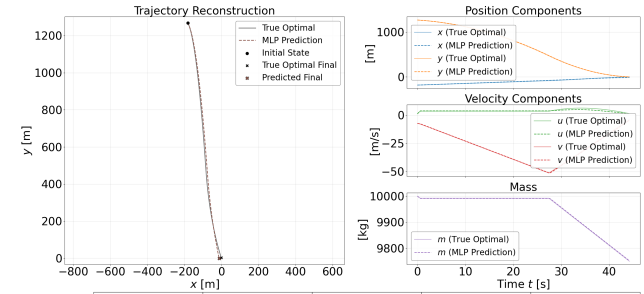


Fig. 4. Visualization of the MLP training dataset showcasing the different optimal landing trajectories.

Once trained, the MLP provides solutions within tens of millisecond-level latency, drastically reducing computational cost and eliminating convergence issues. A fallback strategy is implemented to handle out-of-distribution conditions. Specifically, when the system detects that the initial conditions lie outside the distribution represented in the training data, the control framework switches to a deterministic backup solution. This fallback relies on a classical, computationally efficient polynomial guidance law [18]. While this approach does not guarantee optimality, it ensures safe and predictable system behavior, prioritizing constraint satisfaction and mission safety. An example of MLP performance is shown in Figure 5. The trajectory generation is performed in a reduced 2D manifold defined by the current position and target. The resulting trajectory is then transformed into 3D description and used as a reference for the tracking controller.



	x	y	u	v	m
MLP − True	-11.14 m	-3.68 m	-0.17 m/s	-0.35 m/s	+1.78 kg

Fig. 5. Example of the MLP trajectory reconstruction capability against a true optimal solution.

V. Trajectory Tracking

The final stage of the pipeline is a Model Predictive Control (MPC) layer, which tracks the MLP-derived trajectory within a full 3D dynamical framework [19]. MPC operates by solving a finite-horizon optimal control problem in a receding horizon fashion. The cost function consists of a running term that penalizes tracking errors, defined as the deviation between the predicted state and the reference trajectory along the horizon, and a terminal term that enforces soft landing conditions on position, velocity, and attitude, ensuring near-zero touchdown velocity and proper vertical alignment. The MPC compensates for model mismatches, disturbances, and unmodeled dynamics not captured by the MLP, while enforcing actuator and system constraints. Overall, this hierarchical architecture combines the efficiency of learning-based guidance with the robustness of model-based control, enabling reliable and constraint-aware trajectory execution.

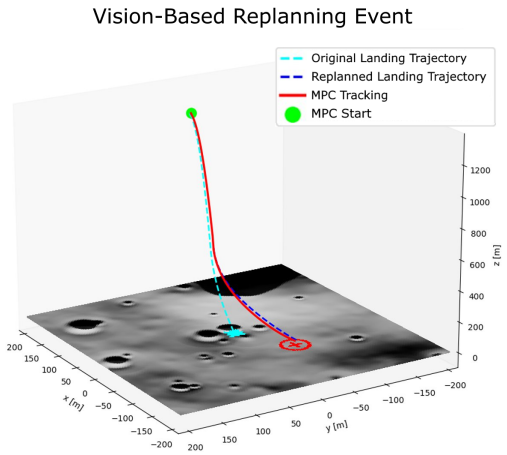


Fig. 6. Integration of the HDA pipeline: the HD module selects a new landing site, the HA module replans the trajectory (cyan to blue), and the MPC tracks the updated reference (red), guiding the lander to the final target.

VI. Discussion and Conclusions

Preliminary results demonstrate the effectiveness of the proposed approach, as shown in Figure 6. The figure illustrates the interaction between the different modules within the proposed pipeline. Following the HD module identification of a new safe landing site (from cyan star to red cross), the HA module updates the nominal trajectory (cyan), redirecting it toward the new target. The MPC layer tracks the references, producing the executed trajectory (red). These results highlight the proposed pipeline as a promising proof of concept, suitable for future validation in hardware-in-the-loop environments and potential deployment in real mission scenarios.

In conclusion, the presented hybrid AI-assisted HDA framework represents a significant advancement toward reliable and explainable autonomous lunar landing. By combining deep learning, uncertainty quantification, and robust control, the proposed system overcomes key limitations of existing approaches, enabling high-precision, real-time landing in challenging lunar environments.

REFERENCES

[1] T. Brady, E. Robertson, C. Epp, S. Paschall, and D. Zimpfer, "Hazard Detection Methods for Lunar Landing," in *IEEE Aerospace Conference*, 2009. doi: 10.1109/AERO.2009.4839354.

[2] A. E. Johnson, A. Huertas, R. Werner, and J. Montgomery, "Analysis of On-Board Hazard Detection and Avoidance for Safe Lunar Landing," in *IEEE Aerospace Conference*, 2008.

[3] J. Schonfeld, "Summary of the Contracted Deliveries of NASA Payloads to the Moon via Commercial Lunar Payload Services (CLPS)," in *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 863–866, 2023.

[4] K. Tomita and K. Ho, "Stochastic Hazard Detection for Landing Under Topographic Uncertainty," arXiv:2305.04249, 2023.

[5] C. D. Epp, T. B. Smith, and E. A. Robertson, "Autonomous Precision Landing and Hazard Detection and Avoidance Technology (ALHAT)," in *IEEE Aerospace Conference*, 2007.

[6] F. Capolupo and A. Rinalducci, "Descent & Landing Trajectory and Guidance Algorithms with Divert Capabilities for Moon Landing," in Proc. AIAA SciTech Forum, 2024, Paper 2024-0086. doi: 10.2514/6.2024-0086.

[7] 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.

[8] L. Ghilardi and R. Furfaro, "Image-Based Lunar Hazard Detection in Low Illumination Simulated Conditions via Vision Transformers," *Sensors*, vol. 23, no. 18, 2023.

[9] 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.

[10] T. Ivanov, A. Huertas, and J. M. Carson, "Probabilistic Hazard Detection for Autonomous Safe Landing," in *2013 AIAA Guidance, Navigation, and Control (GNC) Conference*, 2013.

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

[12] 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.

[13] K. Tomita, K. A. Skinner, and K. Ho, "Bayesian Deep Learning for Segmentation for Autonomous Safe Planetary Landing," *Journal of Spacecraft and Rockets*, vol. 59, no. 6, pp. 1800–1808, 2022.

[14] A. N. Angelopoulos and S. Bates, "A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification," arXiv preprint arXiv:2107.00363, 2022.

[15] Y. Jin, Y. Qu, X. Tong, H. Xie, C. Xiao, Y. Feng, S. Liu, and X. Xu, "Probabilistic Safe Landing Site Selection Based on Estimated Slope and Roughness Error," Transactions in GIS, vol. 29, no. 6, 2025. doi: 10.1111/tgis.70115.

[16] E. Wong, J. Masciarelli, and G. Singh, "Autonomous Guidance and Control Design for Hazard Avoidance and Safe Landing on Mars," in *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, 2002, p. 4619.

[17] C. Sanchez-Sanchez and D. Izzo, "Real-Time Optimal Control via Deep Neural Networks: Study on Landing Problems," Journal of Guidance, Control, and Dynamics, vol. 41, no. 5, pp. 1122–1135, 2018. doi: 10.2514/1.G002357.

[18] S. Ploen, B. Acikmese, and A. Wolf, "A Comparison of Powered Descent Guidance Laws for Mars Pinpoint Landing," in *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, 2006, p. 6676.

[19] T. Inoue, S. Keshtkar, and H. Kojima, "Lunar lander trajectory control using model predictive on–off thruster control," Advances in Space Research, 2026. doi: 10.1016/j.asr.2026.02.069.