Large scale realistic virtual environments for lunar robotics testing using real-time computer games engines

P. Ludivig[1,2] V. Holger[2], J Lamamy[1], [1]ispace (p-ludivig@ispace-inc.com), [2]University of Luxembourg.

Introduction: Virtual simulation environments are a great tool for testing different rover systems before they are sent to the moon. They allow for perfect repeatability and give a better idea how specific parameters can impact the overall system. Most current simulators however suffer from low visual fidelity which is problematic for testing the vision sensors needed for autonomous navigation. Additionally, due to technical and practical limitations, most simulations are limited in environment scale. This is problematic for long range navigation testing needed for missions such as the mars sample fetch rover [1].

In order to solve both issues, we turn towards a popular computer game engine, Unreal Engine 4. It allows for larger environments than what is currently possible in robotics simulators such as Gazebo [2] or V-rep [3]. It also addresses the visual fidelity with a range of tools including real-time raytracing. The tradeoff is a low physics fidelity, which can be an issue when testing wheel-soil interaction. For our use-case we focus primarily on perception systems needed for rover navigations, such as mono and stereo camera systems, where visual fidelity is more important.

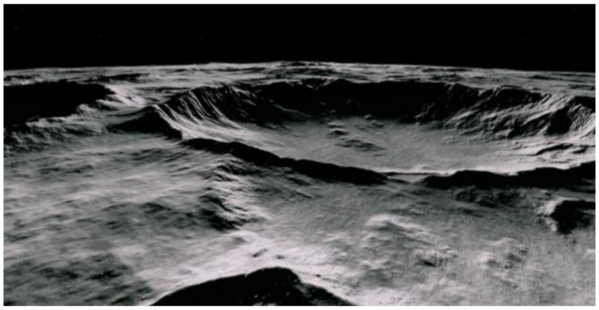

Figure 1. Example of the scale of the lunar simulation environment

Integration: To make testing as flexible as possible, we integrated the Unreal Engine with our existing ROS systems using AirSim [4]. This allows us to use our current pipeline in the same way we interact with all other sensors. As data flows both ways, the ROS environment can also send control command to operate the rover. Due to the distributed nature of ROS, we can also directly transmit the data over the local network from the simulating computer, to the rover processing for hardware in the loop testing.

Large scale environment: To set up a realistic environment in a reasonable time, the simplest approach is to rely on existing environments build for computer games. In our case we opted to modify an

Figure 1. Example image from Unreal Engine 4 lunar simulation environment.

existing lunar environment with a surface area of 64 km^2 [5][6]. The terrain comes with different surface texture and rock distributions. The procedural nature of the surface ensures a variety of conditions thought out the environment. In a second step, we investigated using real data to improve the realism of the environment. We import a DEM based on data from the lunar reconnaissance orbiter (LRO) and add additional detail on top of it, to increase the 1m/px resolution. We rely on Spice [7] to provide correct ephemeris data for both sun and earth positions in the sky.

Methodology: Besides a qualitative comparison approach, we also propose a quantitative approach where we compare the localization accuracy of a visual odometry system for both data from a real camera sensor and a virtual one.

Conclusion: In this paper, we show that a commercially available computer game engine can be used to improve the visual fidelity of simulation environments for planetary robotics. While there are limitations on environment details and sensor noise, we believe that the benefits of environment scale, repeatability and flexibility make this a strong tool for testing perception-based navigation systems.

Acknowledgments: This research was completed with the support of ispace Europe and the University of Luxembourg and was funded by the Luxembourg National Research Fund (FNR grant number 11824057).

References: [1] L. Duvet et al. (2018) 2nd Int Mars Sample Return [2] M. Alan et al. (2019) IEEE Aerospace Conference Proceedings [3] www.coppeliarobotics.com [4] Shah, S et al. (2018) FSR [5] Wu, B. et al. (2019) IEEE IROS, 3262. [6] www.unrealengine.com/marketplace/the-moon [7] https://naif.jpl.nasa.gov/naif/toolkit.html