

# High Resolution Lunar Terrain Generation and Validation

Anders Pearson, Paolo Torrado, Joshua Smith

December 2025

## Abstract

This report details a computational framework for efficiently generating realistic, high resolution, random lunar terrains and validates their statistical fidelity against real lunar surface data. The methodology is based on statistical models of lunar geology, which are dominated by cratering events, to produce high-fidelity synthetic data for machine learning applications. The code can be found at: <https://github.com/anderspearson206/LunarTerrainGenerator>.

## 1 Lunar Terrain Generation Algorithm

This section details the computational algorithm for generating realistic, random lunar terrains. The methodology is grounded in established models of lunar geology and crater formation, implemented in Python with performance optimizations using libraries such as Numba for parallel processing. Without an atmosphere or bodies of water to erode the surface of the moon, the primary driver of topographic change is meteoroid bombardment and the resulting accumulation of impact craters. The process is divided into four stages: precomputation of crater profiles, simulation of cratering events, topographic diffusion to simulate erosion, and micro-roughness synthesis. The first three steps of this process were first presented in [1], and are adjusted in this work for efficiency. The result is an efficient and statistically accurate generator for lunar terrain for resolutions of up to 1 meter per pixel.

### 1.1 Crater Profile Generation

The first step in lunar terrain generation is to define the crater profile. The topographic profile,  $h(r)$ , where  $r$  is the radius of the crater in meters, is modeled by a piecewise function derived in [1] by examining small fresh craters from Lunar Reconnaissance Orbiter Camera (LROC) data:

$$h(r) = \begin{cases} c(D) + h_r & \text{for } \frac{r}{R} < 0.1 \\ c_3(D) \left(\frac{r}{R}\right)^3 + c_2(D) \left(\frac{r}{R}\right)^2 + c_1(D) \left(\frac{r}{R}\right) + c_0(D) + h_r & \text{for } 0.1 \leq \frac{r}{R} \leq 1 \\ h_r e^{-\alpha(\frac{r}{R}-1)} & \text{for } \frac{r}{R} > 1 \end{cases} \quad (1)$$

where  $h_r$  is the rim height, and  $\alpha$  is a decay constant empirically determined to be 3.6. The coefficients  $c(D)$ ,  $c_0(D)$ ,  $c_1(D)$ ,  $c_2(D)$ , and  $c_3(D)$  are diameter-dependent polynomials derived from observational data and can be found in the appendix of [1]. Given that a single square kilometer of lunar surface accumulates over 100,000 impacts, precomputing and caching crater profiles proved essential for computational efficiency.

### 1.2 Cratering Simulation

The crater population is initialized based on the surface age  $t$  (Ga) and the specified simulation area  $A_{\text{sim}}$  ( $\text{km}^2$ ). First, the cumulative crater density for diameters  $D \geq 1 \text{ km}$  is derived from the lunar chronology curve [4, 3]:

$$N_{\text{cum}}(D \geq 1 \text{ km}) = 5.44 \times 10^{-14} (e^{6.93t} - 1) + 8.38 \times 10^{-4} t \quad (2)$$

This reference density is extrapolated to the minimum diameter of interest ( $D_{\min} = 10 \text{ m}$ ) using the Neukum production function, approximated here by a power law  $N_{\text{cum}}(D) \propto D^{-b}$  with  $b \approx 3.35$ . The total number of craters  $M$  to be generated is then calculated as:

$$M = \lfloor A_{\text{sim}} \times N_{\text{cum}}(D \geq 1 \text{ km}) \times D_{\min}^{-b} \rfloor \quad (3)$$

where  $D_{\min}$  is expressed in km. Finally, the individual diameters  $D_i$  for  $i = 1, \dots, M$  are sampled via the inverse transform method [5]:

$$D_i = \left( r_i \frac{N_{\text{cum}}(D \geq D_{\min})}{N_{\text{cum}}(D \geq 1 \text{ km})} \right)^{-1/b} \quad (4)$$

where  $r_i$  is a random variable uniformly distributed in  $[0, 1]$ . A more in-depth derivation of this procedure can be found in [1].

### 1.3 Topographic Diffusion

After the cratering simulation, the terrain undergoes a diffusion process to model the effects of long-term erosion and degradation. This is achieved by solving the heat diffusion equation [2]:

$$\frac{\partial h}{\partial t} = \kappa \nabla^2 h \quad (5)$$

where  $h$  is the surface elevation,  $t$  is time, and  $\kappa$  is the diffusivity constant. This equation is solved numerically using a finite-difference method. The diffusion process smooths sharp features, creating more realistic, aged terrain.

### 1.4 Micro-Roughness Synthesis and Complete Process

To further enhance realism at the target resolution of 1 m/px, the algorithm introduces high-frequency micro-roughness. This includes adding **Perlin noise**, simulating **rocks and boulders** as polygonal protrusions on the heightmap, and superimposing **small, fresh craters**. These final touches ensure that necessary high-frequency spatial details are present in the training data.

The complete process to synthesize a lunar terrain of area  $A_{\text{sim}}$  ( $\text{km}^2$ ) corresponding to a surface age  $t$  is defined as follows:

1. **Pre-computation:** Precompute the crater topographic profiles for the full range of diameters of interest using Equation 1.
2. **Initialization:** Determine the total crater population and size-frequency distribution based on the specified surface age  $t$ , utilizing Equations 2-4.
3. **Time-Stepping Simulation:** Discretize the total geological timeline into intervals of  $\Delta t$ . For each time step:
  - (a) Simulate the crater impacts occurring within this interval by sampling from the distribution and superposing their elevations onto the heightmap.
  - (b) Apply topographic diffusion (erosion) to the global heightmap for the duration  $\Delta t$  using Equation 5
4. **Completion:** Repeat the time-stepping process until the total surface age  $t$  is reached.
5. **Post-Processing:** Apply the micro-roughness elements (noise, rocks, and small craters) to the final eroded terrain.

## 2 Validation against LROC Data

While we based our work off of models derived from real data, it is important to show that these synthetic terrains accurately mimic real lunar terrain. While previous works such as [1] validated terrain generation at resolutions of 2 m/px to 10 m/px, our application requires high-fidelity data at 1 m/px. We incorporated micro-roughness elements (Perlin noise and rock distribution) and validated the output in two stages: first against a broad dataset of standard LROC Narrow Angle Camera (NAC) Digital Terrain Models (DTMs) (at resolution of 2 m/px), and second against one rare high-resolution 1 m/px DTM.

## 2.1 Methodology

We utilized five key topographic statistics defined in [1] to evaluate terrain similarity: Bidirectional Slope, RMS Height, Hurst Exponent, Absolute Slope, and Differential Slope.

For the broad validation, we selected 17 LROC NAC DTMs representing diverse lunar geographies at 2 m/px resolution, including Apollo landing sites, cratered highlands (e.g., Tycho), and maria regions. These large DTMs were divided into  $256 \times 256$  m patches, which resulted in a dataset of over 60,000 real terrain patches for comparison against our synthetic generator.

For the high-resolution validation, we compared our synthetic output against the LROC NAC Haworth DTM, which offers native 1 m/px resolution, and is one of the lunar surface DTMs available at 1 m/px resolution. It is again divided into  $256 \times 256$  m patches for comparison to our dataset.

We compare the real data to our dataset of 8,000  $256 \times 256$  synthetic lunar terrains at 1 m/px resolution.

## 2.2 Statistical Comparison Results

### 2.2.1 Validation at 2 m/px Resolution

We first compared the statistical distribution of the synthetic terrains against the aggregate of the 17 LROC DTMs at 2 m/px. The results show a strong similarity across all statistics and are defined in Table 1

Table 1: Statistical Comparison at 2 m/px (Synthetic vs. 17 LROC DTMs)

Metric	Synthetic (Median [IQR])	Real (Median [IQR])
Bidirectional Slope ( $^{\circ}$ )	7.40 [4.09 – 12.40]	7.90 [4.40 – 13.11]
RMS Height (m)	1.70 [0.90 – 3.09]	1.90 [1.01 – 3.16]
Hurst Exponent	0.90 [0.88 – 0.92]	0.88 [0.85 – 0.89]
Breakover Point (m)	10.0 [9.0 – 10.0]	10.0 [7.0 – 10.0]
Absolute Slope ( $^{\circ}$ )	1.52 [0.83 – 2.66]	1.64 [0.90 – 2.84]
Differential Slope ( $^{\circ}$ )	0.84 [0.39 – 1.61]	0.96 [0.39 – 2.55]

### 2.2.2 Validation at 1 m/px Resolution

To validate the efficacy of our micro-roughness synthesis, we compared the generator outputs against the 1 m/px Haworth DTM. Again, the results shown in Table 2 support the accuracy of the synthetic terrain.

Table 2: Statistical Comparison at 1 m/px (Synthetic vs. Haworth DTM)

Metric	Synthetic (Median [IQR])	Real (Median [IQR])
Bidirectional Slope ( $^{\circ}$ )	7.60 [4.18 – 13.10]	7.99 [5.08 – 12.07]
RMS Height (m)	0.90 [0.49 – 1.61]	0.99 [0.65 – 1.49]
Hurst Exponent	0.898 [0.894 – 0.903]	0.898 [0.893 – 0.904]
Breakover Point (m)	13.0 [11.0 – 15.0]	12.0 [10.0 – 13.0]
Absolute Slope ( $^{\circ}$ )	1.53 [0.84 – 2.67]	1.61 [1.03 – 2.46]
Differential Slope ( $^{\circ}$ )	0.57 [0.26 – 1.01]	0.34 [0.12 – 0.74]

The close correspondence in slope statistics and fractal characteristics validates that the synthetic datasets generated by this pipeline are high-fidelity approximations of the lunar environment, suitable for training physics-informed neural networks.

## References

- [1] Yuzhen Cai and Wenzhe Fa. Meter-scale topographic roughness of the moon: The effect of small impact craters. *Journal of Geophysical Research: Planets*, 125(8):e2020JE006429, 2020.

- [2] Caleb I Fassett and Bradley J Thomson. Topography of the moon’s southernmost polar regions. *Journal of Geophysical Research: Planets*, 119(10):2255–2273, 2014.
- [3] Harald Hiesinger, Carolyn H van der Bogert, and Nico Schmedemann. Chronology and evolution of the lunar highlands: A new paradigm. In *Recent Advances and Current Research Issues in Lunar Stratigraphy*, volume 482, pages 1–52. Geological Society of America, 2011.
- [4] Gerhard Neukum, Boris A Ivanov, and William K Hartmann. Cratering records in the inner solar system in relation to the lunar reference system. *Space Science Reviews*, 96(1-4):55–86, 2001.
- [5] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 3rd edition, 2007.