USING SIMULATED LUNAR IMAGERY TO TRAIN REAL NETWORKS

Kyle McCleary[1*], Haochen Zhang[1], Paulo R.M. Fisch[1], Zachary Manchester[1], and Brandon Lucia[1]; [1]Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, [*][kmcclear@andrew.cmu.edu]

Abstract. *Compact CPUs and GPUs enable new on-satellite capabilities previously relegated to ground stations. One such application is optical navigation using landmarks, such as features on the Moon's surface. Advances in machine learning enable complex image processing tasks, but deep neural networks require significant imagery for training. The goal of this extended abstract is to demonstrate the viability of training a neural network on synthetic imagery to detect landmarks in real Moon images. Experimentation shows that this is a viable approach to solving the data problem for training neural networks for Moon image processing.*

Introduction. Identifying landmarks in lunar imagery is useful for optical navigation and registration of captured images, but current methods perform poorly when faced with off-nadir viewing angles or large search areas.[1] When trained properly, convolutional neural networks (CNNs) are robust to variations in image characteristics such as lighting and view angle, and have been used previously for landmark detection for satellite orbit determination of an Earth-based satellite.[2] The caveat is that improving CNN robustness requires a large and accurately labeled dataset containing many variations of views, lighting conditions, and camera characteristics. Using synthetic imagery would allow for all of these problems to be solved, but neural networks have previously struggled to generalize from synthetic to real imagery.

The goal of this work is to demonstrate the viability of using only synthetic imagery of the Moon made with open source tools to train an object detection network to detect landmarks in real Moon imagery.

Dataset Generation. This section describes the generation of the dataset used to train and evaluate the object detection network. First, landmarks on the surface of the Moon are selected using a similar process to VINSat.[2] Next, synthetic images are generated from a 3D model of the Moon and labeled using the pre-selected landmark locations to create a novel dataset. Real Lunar Reconnaissance Orbiter Camera (LROC) mosaics are downloaded and labeled with the same landmark location list. The training dataset contains 10,000 synthetic images for training and 1000 LROC mosaic images for evaluation.

Synthetic Moon Model. The 3D model of the Moon is built in Blender, an open-source 3D rendering software.[3] The model is built from a sphere with a radius of 1737.4 km with a digital elevation map (DEM) used to offset the surface of the sphere.[4] A color map of the Moon built from Hapke parameter maps built from LROC collections is overlaid on the model.[5] The model additionally contains a "Sun" lighting source and a camera. Due to memory constraints, only the portions of the DEM in view of the camera are dynamically loaded for each image render.

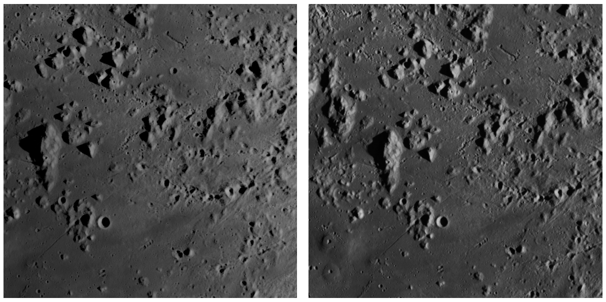
Generating and Labeling Synthetic Images. The synthetic dataset is built using an iterative process. First, the predefined landmarks over a specific Moon region are added to the scene as non-visible objects. Next, a random camera position between 50 and 150 km altitude is chosen over the current Moon region. The Sun is rotated randomly between $\pm 90^\circ$ of the spacecrafts longitude and between $\pm 1.59^\circ$ latitude. Next a random Sun intensity is applied. The regions of the Moon's surface in view of the camera are determined and used to load the appropriate missing DEM data. The landmarks in view of the camera and their bounding boxes are calculated and saved to a text label file. The image is then rendered and saved as a PNG file. To mimic LROC Wide Angle Camera (WAC) imagery as if captured by a typical staring array camera, imagery is captured at a 704x704 resolution, matching the height of the simulated image to the scan-width of the WAC color imagery. The simulated camera has a field of view of 60° to match WAC color imagery. This iterative process is outlined in Algorithm 1 and an example synthetic image can be seen in Figure 1 on the left.

Algorithm 1 Generating and Labeling Synthetic Images

Require: num_images ≥ 1
  Set scene constraints
  Load landmarks into scene
  Initialize camera parameters
  $N \leftarrow$ num_images
  while $N \geq 1$ **do**
  1. Randomize camera placement
  2. Randomize Sun orientation
  3. Randomize Sun intensity
  4. Load missing DEM data
  5. Remove unneeded DEM data
  6. Render image
  7. Create label file

  end while

Processing and Labeling Real Moon Image Data. The real imagery dataset is created in a similar way to the synthetic dataset, but with some key differences. A mosaic of real Moon imagery is overlaid on the digital surface in the Blender model. The camera is again randomly placed, but the Sun orientation is placed directly overhead due to the lunar imagery already containing the appropriate shading.



(a) Synthetic Image Data (b) Real Image Data

Figure 1. Synthetic image data and real LROC image data comparison.

An example images made from real LROC WAC data can be seen on the right in Figure 1.

Landmark Detection Network. The landmark detection network is a standard YOLOv8m[6] object detection network trained on synthetic images and validated using real LROC images. While there may be better alternatives for this task, the authors' familiarity with this framework allowed for rapid prototyping to determine if further exploration is warranted.

Evaluation and Results. This section discusses the evaluation of the landmark detection network and the results.

The landmark detection network training and validation results were recorded over time. The details shown in Fig. 2, demonstrate the ability of the network to improve performance on real data while training only on synthetic data. Over 30 epochs, the network achieves a mAP50 of 0.824 and an F1 score of 0.74 at 0.192 confidence when evaluated on real imagery. An example real image with detections from the network is shown in Fig. 3.

Conclusions and Future Work. While there are many avenues for improvement, this work has demonstrated the viability of training a CNN on only synthetic imagery to perform landmark detection on real Moon images. Some of the many options to explore include: increasing the network size, improving labeling, improving landmark selection, and changing the network setup entirely.

Using a synthetic Moon model that we open source, we can simulate virtually any lighting condition, location, and camera model for training CNNs. These CNNs can then be used directly on-orbit in future lunar missions, opening the door for many new on-orbit applications including optical navigation.

The following page contains a variety of additional examples of real and synthetic lunar images.

Disclaimer. The views expressed are those of the author and do not necessarily reflect the official policy or position of the Department of the Air Force, the Department of Defense, or the U.S. government.

References.

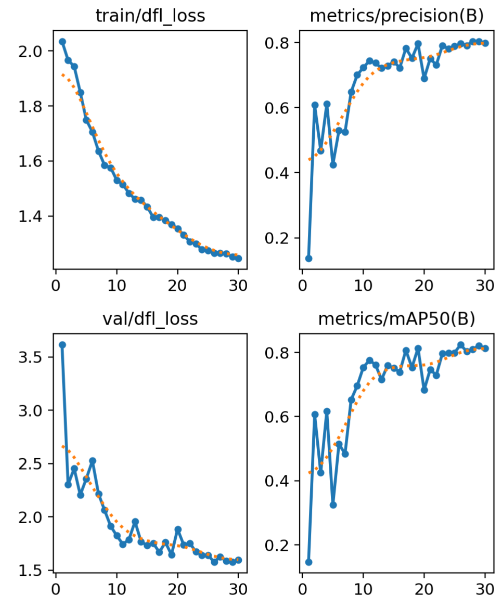


Figure 2. Training and validation metrics during network training process.

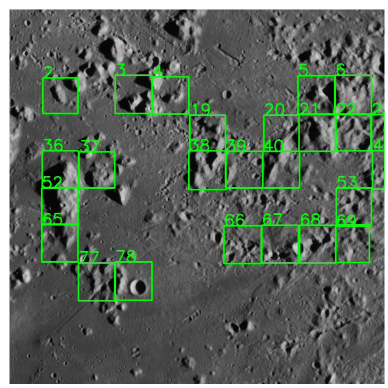


Figure 3. Real image with detections from network.

[1] J. A. Christian et al., "Lunar Crater Identification in Digital Images," *The Journal of the Astronautical Sciences*, vol. 68, pp. 1056–1144, Dec. 2021. doi: 10.1007/s40295-021-00287-8.

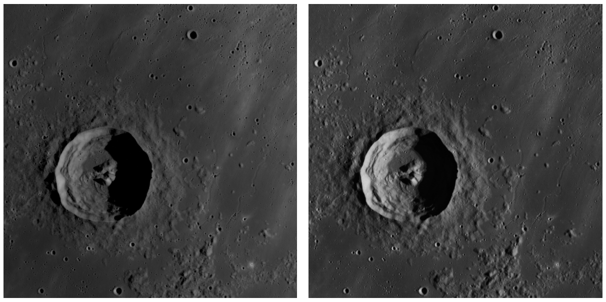
[2] K. McCleary et al., "VINSat: Solving the Lost-in-Space Problem with Visual-Inertial Navigation," *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[3] Blender Online Community, *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018.

[4] M. K. Barker et al., "A new lunar digital elevation model from the lunar orbiter laser altimeter and selene terrain camera," *Icarus*, vol. 273, pp. 346–355.

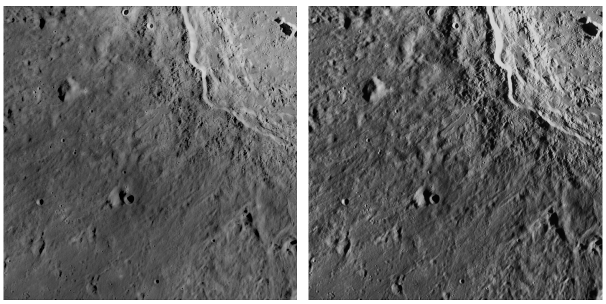
[5] H. S. et al., "Lunar mare tio2 abundances estimated from uv/vis reflectance," *Icarus*, vol. 296, pp. 216–238, 2017.

[6] G. Jocher et al., "YOLO by ultralytics," Jan. 2023.



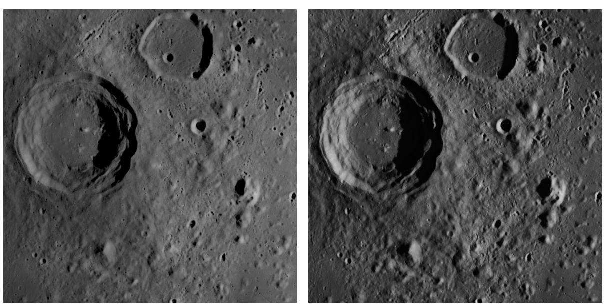
(a) Synthetic Image Data (b) Real Image Data

Figure 4. Synthetic vs Real Comparison



(a) Synthetic Image Data (b) Real Image Data

Figure 5. Synthetic vs Real Comparison

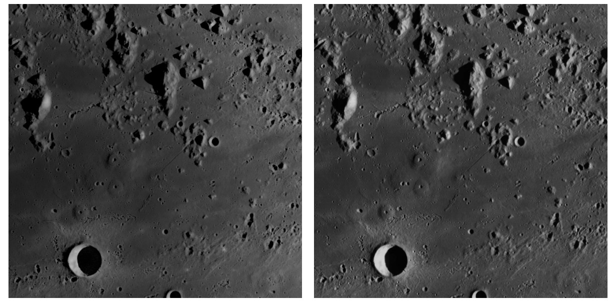


(a) Synthetic Image Data (b) Real Image Data

Figure 6. Synthetic vs Real Comparison

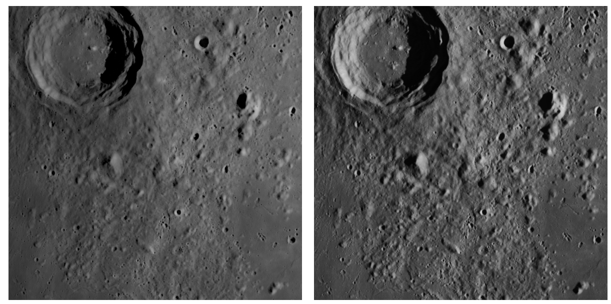
(a) Synthetic Image Data (b) Real Image Data

Figure 7. Synthetic vs Real Comparison



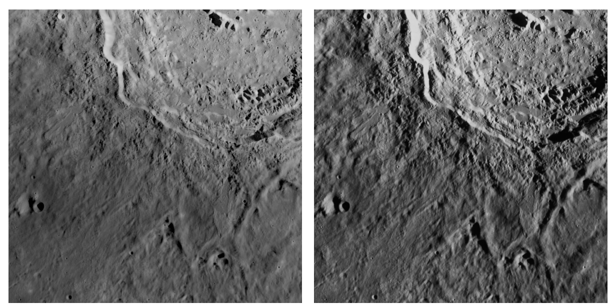
(a) Synthetic Image Data (b) Real Image Data

Figure 8. Synthetic vs Real Comparison



(a) Synthetic Image Data (b) Real Image Data

Figure 9. Synthetic vs Real Comparison



(a) Synthetic Image Data (b) Real Image Data

Figure 10. Synthetic vs Real Comparison

(a) Synthetic Image Data (b) Real Image Data

Figure 11. Synthetic vs Real Comparison