Deep Learning Crater Detection for Lunar Terrain Relative Navigation

Deep Learning Crater Detection for Lunar Terrain Relative Navigation

Lena M. Downes*
Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts, 02139

Ted J. Steiner†
Draper, 555 Technology Square, Cambridge, Massachusetts, 02139

Jonathan P. How‡
Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts, 02139

Terrain relative navigation can improve the precision of a spacecraft's position estimate by providing supplementary measurements to correct for drift in the inertial navigation system. This paper presents a system, LunaNet, that uses a convolutional neural network to detect craters from camera imagery taken by an onboard camera. These detections are matched with known lunar craters, and these matches can be used as landmarks for localization. The motivation for generating such landmarks is to provide relative location measurements to a navigation filter, however the details of such a navigation filter are not explored within this work. Our results show that on average LunaNet detects approximately twice the number of craters in an intensity image as two other intensity image-based crater detectors. One of the challenges of cameras is that they can generate imagery with vastly different appearances depending on image qualities and noise levels. Differences in image qualities and noise levels can occur for reasons such as changes in irradiance of the lunar surface, heating of camera electronic elements, or the inherent fluctuation of discrete photons. These image noise effects are difficult to compensate for, making it important for a crater detection system to be robust to them. Convolutional neural networks have been demonstrated to be robust to these kinds of imagery variation. LunaNet is shown to be robust to four types of image manipulation that result in changes to image qualities and noise levels of the input imagery.

I. Introduction

Previous lunar landings relied mostly on inertial navigation methods, which time-integrate measurements from an inertial measurement unit (IMU). However, inertial navigation systems accumulate error over time and past lunar landing errors have been on the order of up to several kilometers [1]. These errors pose a challenge because many of the locations of interest for future landing missions are in or near hazardous terrain, which motivates the need for increased landing precision.

One solution for reducing landing error is terrain relative navigation (TRN), which uses sensors to observe the terrain around a vehicle in order to estimate the vehicle's position. Vision-based TRN systems are especially attractive due to the small size, weight, and power requirements of space-rated cameras, as well as the rich data they provide. Craters are a type of visual landmark in the terrain that are visible in a variety of lighting conditions from most locations on the Moon and with a variety of cameras. Current crater databases cover approximately 80% of the nearside of the Moon [2].

Our system, LunaNet, visually detects craters in a camera image using a neural network and matches those detected craters to a database of known lunar craters with known latitudes and longitudes. The inclusion of these measurements in a landing navigation system can reduce estimation error and enable increased-precision navigation both in-orbit and during landing [3]. However, camera imagery is prone to variation caused by many sources, including time of day, electronic component heating, and type of camera. The resultant image effects are difficult to predict with high certainty and compensate for, making it important that a crater detector be robust to such changes, as LunaNet is.

*Draper Fellow, S.M. Candidate, Aeronautics and Astronautics, MIT, lmdownes@mit.edu.
†Senior Member of the Technical Staff, Perception and Autonomy, Draper, tsteiner@draper.com.
‡Richard C. Maclaurin Professor of Aeronautics and Astronautics, Aeronautics and Astronautics, MIT, jhow@mit.edu, AIAA Fellow.

Developments in deep learning for computer vision have resulted in significant advances in the capabilities of vision-based systems. However, the space industry has been reluctant to adopt deep learning-based systems due to concerns over reliability [4]. Despite these concerns, our results show potential for robustness and reliability improvements, as well as better generalization and reduced testing and modeling burden. LunaNet demonstrates the potential for a neural network to replace a small piece of a larger system, thereby improving performance and robustness to image noise while maintaining a restricted domain of failure modes.

II. Related Work

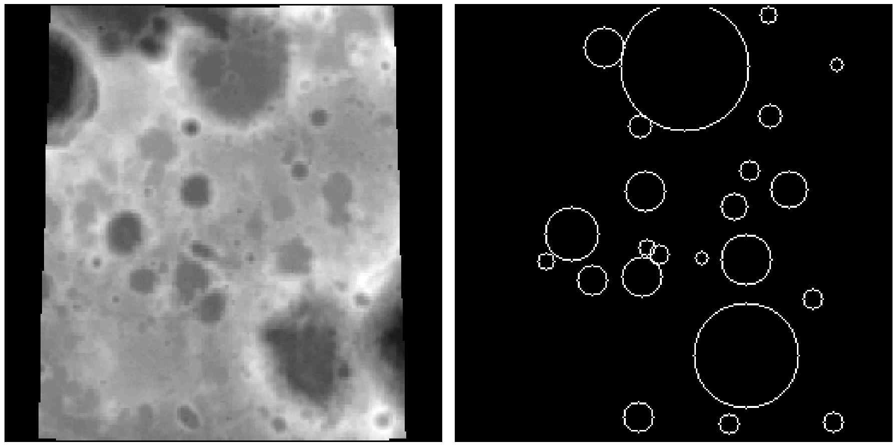
Terrain relative navigation has been studied for years as a method to improve the landing accuracy for lunar landers [5–7]. A common theme in lunar TRN literature is to focus on detecting craters and using them as landmarks for localization because craters are present on nearly the entire lunar surface, many of their locations have already been catalogued, and crater information is efficient to store [3]. In addition, a crater's location can be uniquely identified based on the local constellation of nearby craters.

Previous crater-based TRN systems utilized traditional image processing techniques to detect craters from imagery. However, these approaches are typically not robust to changes in lighting conditions, viewing angles, and image noise. The system of [3] is a lunar terrain relative navigation system which uses image processing methods to visually detect craters. To detect craters, it thresholds the input grayscale intensity image for bright patches and for dark patches, which correspond to crater rims and shadows. Edges of the patches are detected, and pairs of dark and bright edges are fitted with ellipses, which represent crater rims. Due to its thresholding process, we refer to this crater detector as a trinary edge detector. This work demonstrated the potential for craters to improve the position tracking of a spacecraft in lunar orbit. However, the trinary edge detector struggles to consistently detect the same craters from frame to frame in a trajectory and experiences significant sensitivity to image quality, brightness, and shadowing. This makes precisely tuning the trinary edge detector critical to its performance. Precise tuning requires high levels of human input and detailed modeling, and is only possible if the camera imagery qualities and noise levels are known with high confidence prior to the mission. It is often difficult to predict these qualities, and they are not necessarily static during a mission.

Recent work has demonstrated that significant improvements in robustness to lighting effects for automated crater detection are possible by applying advances from the fields of computer vision and deep learning. PyCDA [8] is a neural network-based crater detector composed of a detector, extractor, and classifier. These components are combined in sequence to generate crater predictions. The detector takes as input a grayscale intensity image and uses a downsized U-net [9] architecture to generate per-pixel likelihoods that each pixel is a part of a crater rim. This pixel prediction map is input to the extractor, which labels connected components in the prediction map image and fits circles to these components, generating a list of crater candidates. The crater candidates are input to the classifier, which is a convolutional neural network (CNN) with a 12 by 12 pixel classifier of 2D convolutions. The classifier makes a binary prediction of whether the crater candidate is a true crater. PyCDA outputs the crater candidates that the classifier determined were true craters. This work focused on detecting new craters which had not been previously catalogued, while the goal of LunaNet is to detect craters and match them to catalogues of known craters, so that they can be used to localize a spacecraft in real-time.

Another recent system, DeepMoon [10], applies a CNN to detect craters from elevation data represented as overhead imagery. It uses a U-net [9] architecture to perform pixel-wise classification of craters. After generating a prediction image of crater rim locations, DeepMoon uses template matching to obtain discrete crater detections from the prediction image. DeepMoon utilizes elevation data in the form of digital elevation map (DEM) imagery, which has significantly different micro-scale variation than camera imagery and is not affected by lighting effects such as glare and shadowing. Obtaining an accurate elevation map in-flight requires a space-rated range sensor such as radar or LiDAR, which is more expensive on average than a space-rated camera. Due to this, LunaNet uses camera images and thus must accommodate for shadows and other forms of visual noise, which introduces additional detection complexities. Like PyCDA, the work of DeepMoon also did not focus on identifying or matching the craters it detected to known craters. Furthermore, in order to provide measurements for the onboard navigation filter, LunaNet must be capable of running in real-time on a space-rated processor, while both PyCDA and DeepMoon were primarily designed to be used for post-processing.

LunaNet's architecture is based on the U-net used by DeepMoon. This is a pixel-wise classifier with a contracting and expanding path. The training images that were input to train DeepMoon were sets of ground-truth classification data (known craters from a crater database) and digital elevation maps (DEM) of a corresponding area on the Moon. The elevation images that DeepMoon was trained on are elevation maps of the Moon's surface, where a darker pixel corresponded to a depression and a lighter pixel corresponded to a raised area, as shown in Fig. 1(a). The ground-truth



(a) Moon digital elevation map (DEM) input used to train DeepMoon.

(b) Ground-truth craters in camera field of view.

Fig. 1 Input image pairs used to train DeepMoon for crater detection from DEM.

classification data were images with black backgrounds, and white rings corresponding to the pixel locations of crater rims, as shown in Fig. 1(b). This ground-truth crater data was obtained from a combination of two human-generated lunar crater databases: the 5-20 km database from [11] and the >20 km database from [12]. DeepMoon was trained for four epochs on 30,000 DEM images.

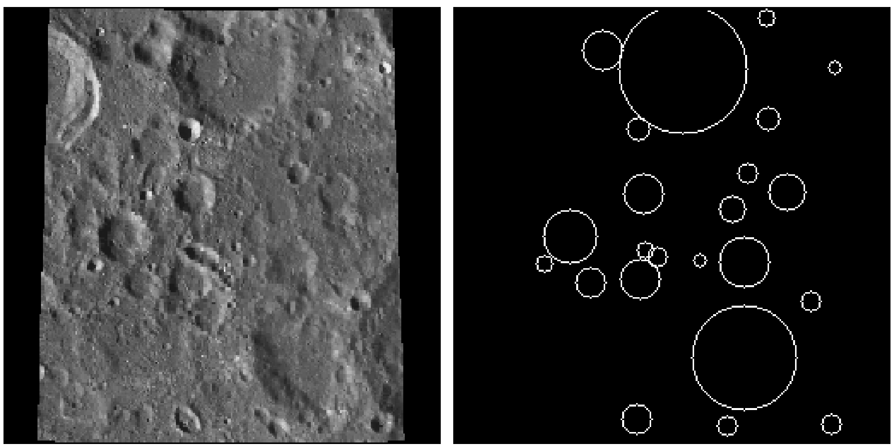
III. Methods

Our method, LunaNet, combines a CNN with image processing to detect craters in lunar surface imagery that is captured by an onboard camera in real-time. The CNN architecture is based on that of [10]. The output of the neural network is a grayscale image with brighter pixels corresponding to predicted crater rims. This output prediction is processed to identify likely craters in the image. These craters are matched against the databases of known craters from [11] and [12] in order to identify their true locations on the surface of the Moon. The detected crater centers in the image and the known 3D locations of the craters can then be passed to a navigation filter to be used as measurements to improve the spacecraft's position and attitude estimates and correct for drift from the inertial sensors.

LunaNet was initialized on the weights from [10] for classification of an elevation image, and was additionally trained with 800 Lunar Reconnaissance Orbiter Camera (LROC) [13–16] intensity images for 10 epochs. Table 1 shows the hyperparameters that were used to complete the additional intensity image training. The training images were obtained from the LROC Wide Angle Camera (WAC) Global Morphology Mosaic with resolution of 100 meters per pixel. This mosaic is a simple cylindrical map projection comprising over 15,000 images from $-90°$ to $90°$ latitude and all longitudes. The mosaic was converted to a 16-bit GeoTiff, and rescaled to an 8-bit PNG file. The input training images were cropped from the mosaic using similar methods to those of [10]: randomly cropping a square area of the mosaic, downsampling the image to 256 x 256 pixels, transforming the image to orthographic projection, and rescaling the image to more closely match the intensity distribution of a DEM image. LunaNet was trained using the ADAM optimizer [17]. The sets of training images that were input to the neural network to complete the additional training were also similar in format to those that were originally input into [10], with the exception that they were intensity images instead of elevation images, as shown in Fig. 2(a).

Table 1 Parameters used to create LunaNet by training [10] with intensity images

Parameter	Value
Learning Rate	10^{-4}
Batch Size	8
Filter Length	3
Weight Regularization	10^{-5}
Number of Filters	112
Dropout Fraction	0.15
Number of Runs	1

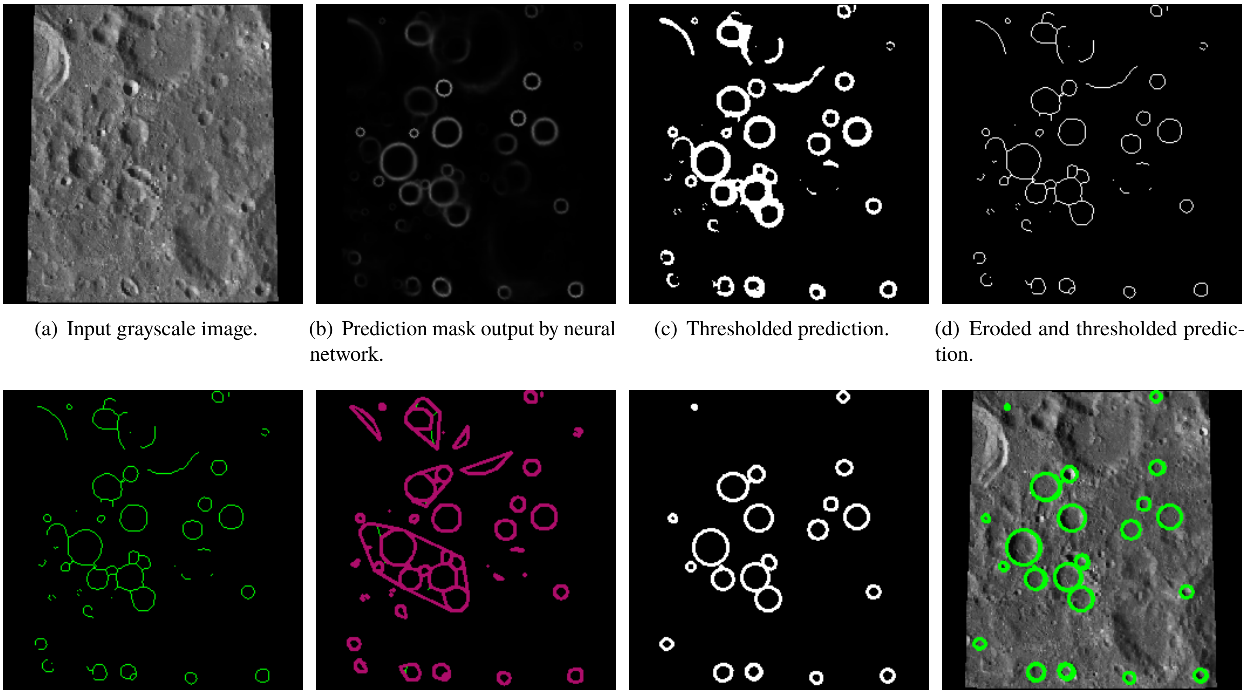


(a) Grayscale intensity image input to LunaNet. (b) Ground-truth craters in camera field of view.

Fig. 2 Input image pairs used to train LunaNet for crater detection from intensity images.

The neural network returns grayscale images with bright pixels corresponding to crater rims predicted from the input image, not discrete crater detections. In order to reduce computation and match crater detections to known craters, it is necessary to obtain a list of discrete crater detections with the pixel locations of their centers and their radii in pixels. For this reason, the prediction images are processed to obtain the discrete detected craters. Fig. 3 shows the following processing steps that are performed on the crater predictions. The prediction masks are thresholded by intensity to obtain predictions with higher levels of detection certainty. The predictions are then eroded to find single-pixel lines of crater rim detections. Following this, the contours of the eroded predictions are found and these are used to eliminate individual detections which are smaller than four pixels, since at least four points are needed to fit an ellipse. Convex hulls are found for these contours, which enables completion of circles for partially detected crater rims. The convex hulls are then fit with ellipses, with a maximum allowable eccentricity of 10%. The ellipses are not allowed to have greater than 10% eccentricity because true craters are generally circular, and detections that are more eccentric than this tend to be false detections. This process produces our final crater detections for the image: discrete, closed, circular ellipses that represent the rims of the detected craters.

A spacecraft location estimate enables the prediction of what craters should be in the camera field of view. To simplify the simulations for this work, the spacecraft location estimate was set equal to the true spacecraft location. In order to test the crater matching performance, we produce a search area that is centered at the estimated spacecraft location and is as wide as the camera field of view. This search area contains all of the known, catalogued craters for that area, with their latitudes, longitudes, and radii. These known craters are projected into the camera pixel space, giving each known crater a size and location in pixels, based on the location estimate.



(a) Input grayscale image.

(b) Prediction mask output by neural network.

(c) Thresholded prediction.

(d) Eroded and thresholded prediction.

(e) Contoured to find connected predictions.

(f) Convex hulls of contours.

(g) Fitted with ellipses to form closed circular crater rim detections.

(h) Final crater predictions overlaid on top of intensity image.

Fig. 3 Processing operations performed on neural network prediction image to obtain final discrete crater detections output by LunaNet.

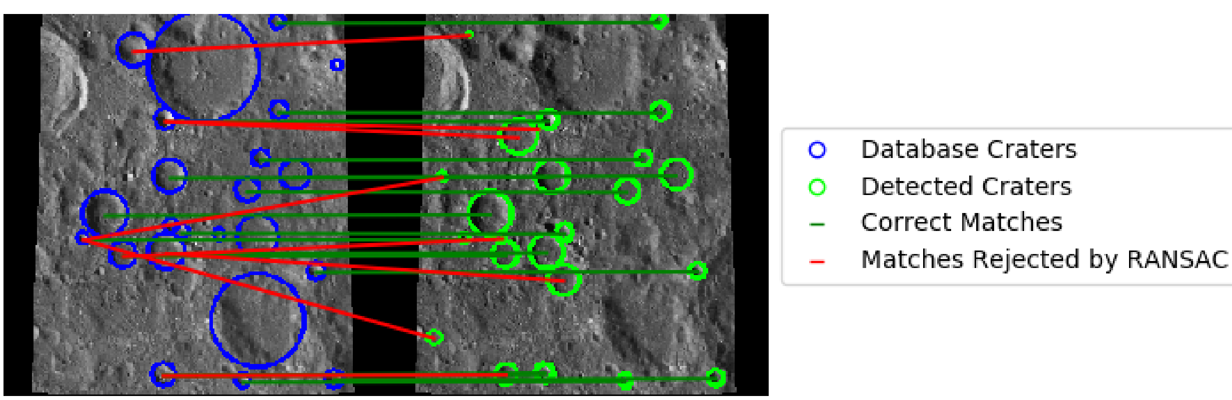
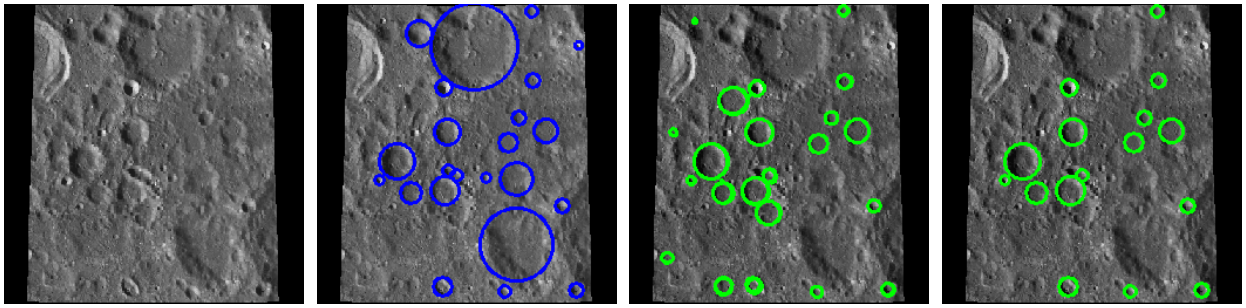


Fig. 4 Matching process of detected crater with database craters. Ground-truth set of craters that appear in the crater database for this test area in blue, craters detected by LunaNet in light green. Inlier pairs marked with dark green lines, outlier pairs marked with red lines.



(a) Grayscale intensity image input to LunaNet. (b) Ground-truth craters in camera field of view. (c) Detected craters. (d) Matched craters.

Fig. 5 Craters that were detected by LunaNet and matched to known craters in a representative LRO image with no noise added.

The detected craters are matched to the known craters by means of nearest neighbor matching. Each detected crater is paired with a known crater that is closest to it in x, y pixel space, as well as in diameter in pixels. These pairs are then processed with random sample consensus (RANSAC) [18] to eliminate outlier pairs. A pair is determined to be an outlier if the the translation vector between the detected crater and the known crater is sufficiently different from the translation vector of all inlier pairs, as seen in Fig. 4. This complete matching process is shown in Fig. 5, with the final accepted matched craters shown in Fig. 5(d).

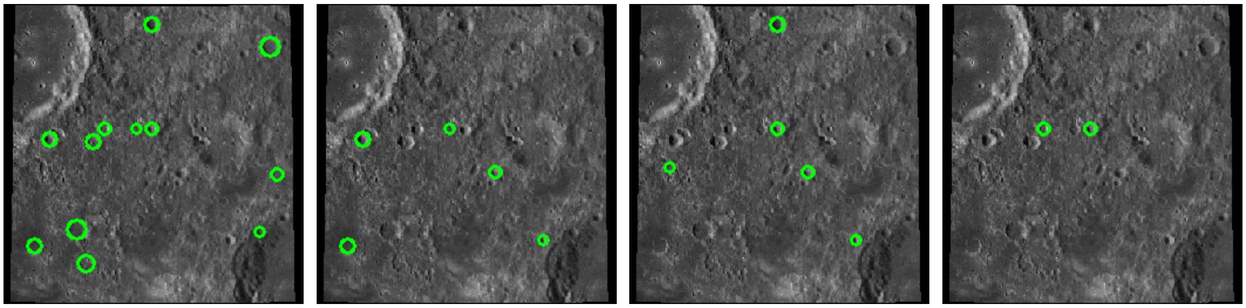
The difference between the matched crater location and the detected crater location generates a measurement of the error in the position estimate. This measurement can be used to improve the spacecraft position estimate by incorporating it into the navigation system, which may include an extended Kalman filter [19]. As the spacecraft moves around the Moon, repeated crater detections and matches can correct for the drift that is typical in inertial navigation systems.

IV. Results

The additional training that was performed on top of DeepMoon to produce LunaNet's CNN and the additional processing of the prediction image provided a substantial improvement to the number of craters detected in an intensity image. DeepMoon was not trained to detect craters in an intensity image; the comparison with DeepMoon serves to demonstrate how LunaNet expands upon DeepMoon's capabilities. LunaNet is also compared against another neural network-based crater detector, PyCDA [8], and the trinary edge detector [3], a crater detector that is based on thresholding and edge-pairing.

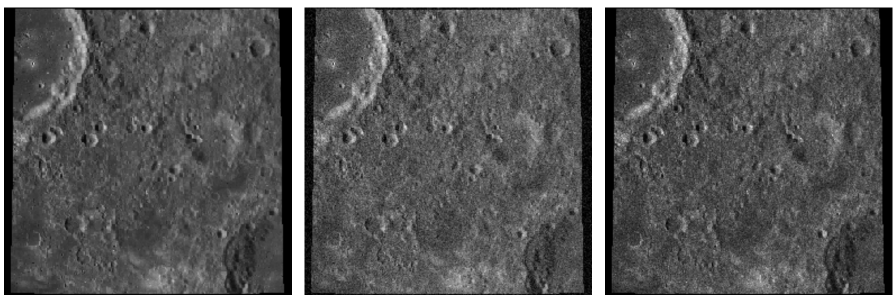
Detecting more craters in an image is important because a higher number of detected features corresponds to lower position estimate uncertainty in terrain relative navigation. Detecting features with high accuracy was also demonstrated to be beneficial for TRN in [20]. Fig. 6 shows the crater detection performance of the four crater detectors on the same image, an LRO image of a region near the lunar equator. These images show craters that were detected and successfully matched to known lunar craters. The threshold levels of the trinary edge detector were tuned to optimally detect craters in this LRO imagery. DeepMoon, PyCDA and the trinary edge detector all detect less than 50% the number of craters that LunaNet detects.

Since LunaNet targets use as a real-time navigation system, it needs to have reliable performance regardless of the appearance of the lunar surface in the camera image. Differences in the appearance of the lunar surface could unknowingly occur due to differences in time of day, exposure of the camera, or albedo properties of the local lunar surface. It is difficult to determine a priori exactly how the lunar surface will appear in a camera image, therefore it is important to design a system that is not extremely sensitive to these image qualities that are difficult to determine with high confidence prior to the mission. To simulate unexpected changes in the imagery, the crater detectors were tested on LRO imagery with a variety of noise and effects. These included imagery with added Gaussian noise, which occurs due to thermal radiation of the camera's electronic components, imagery with shot noise, which occurs due to statistical quantum fluctuations, and imagery with 30% boosted brightness and 30% lowered brightness, which can occur due to variation in the tuning of the exposure adjustment algorithm [21]. Examples of these different kinds of images can be seen in Fig. 7, where the effects were all applied to the same LRO image.

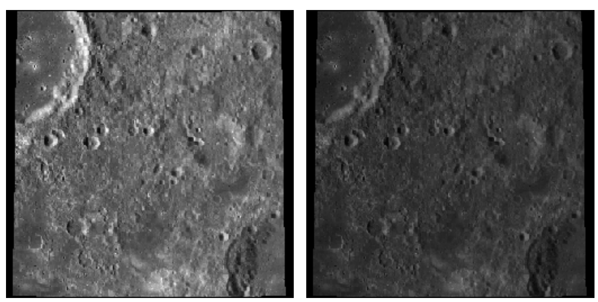


(a) LunaNet (b) DeepMoon (c) PyCDA (d) Trinary edge

Fig. 6 Craters that were detected by different crater detectors and matched to known craters in a representative LRO image with no noise added.



(a) Clean LRO image. (b) Gaussian noise. Mean of 10, standard deviation of 15. (c) Shot noise. Expected separation on interval 0 to 102.



(d) Brightness increased by 30%. (e) Brightness decreased by 30%.

Fig. 7 Comparison of LRO image with and without different kinds of noise.

In order to test the detection performance of LunaNet on imagery with different noise effects, a set of 100 test images were generated from LRO imagery. These test images were randomly cropped from the LRO mosaic, and were centered around the lunar equator to reduce warping effects. Gaussian noise, shot noise, and brightness shifts were applied to the same sequence of images to produce five sets of test images that feature the same geographical area, but with and without various kinds of noise. All four crater detectors were tested on these five sets of imagery to determine the number of good crater detections that each detector produces on an average image. A good crater detection is defined for this purpose as being sufficiently close in pixel space and in diameter size to a known crater such that it can be considered a crater match. For this simulation, the sum of error in crater center and diameter had to be less than ten. This definition of a crater match places value on detecting crater rims accurately; detected craters that do not closely match known crater rims will not be accepted. Importance was placed on crater accuracy because high accuracy features yield good convergence in a navigation system. Since LunaNet is intended to be used for real-time terrain relative navigation it is important for it to identify many accurate craters, but not necessarily for it to identify all of the craters in the image.

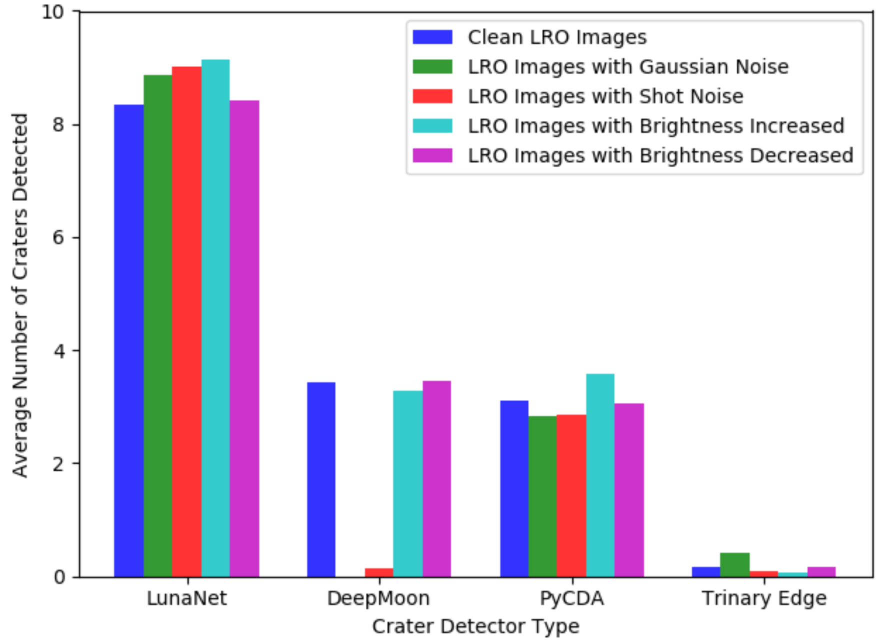


Fig. 8 Comparison of average number of matched crater detections from different crater detectors on different images. Detectors were tested on the same set of 100 LRO grayscale intensity images without noise added, with Gaussian noise added, with shot noise added and with brightness shifts added.

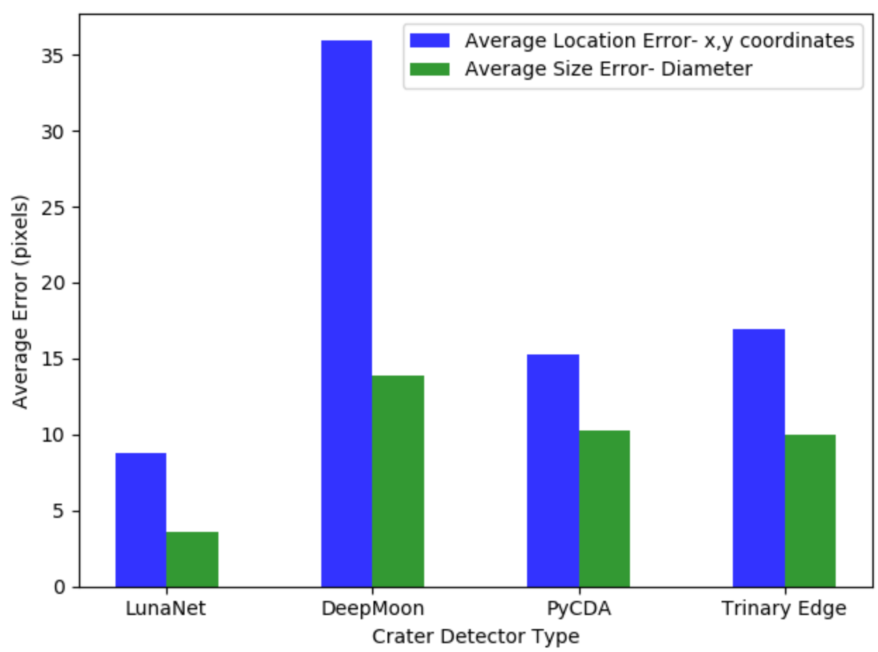


Fig. 9 Comparison of accuracy of crater detections from different crater detectors on different images. Location accuracy is the Euclidean distance error between the detected crater center and the known crater center in pixels. Size accuracy is the error between the detected crater diameter and the known crater diameter in pixels. Detectors were tested on the same set of 100 LRO grayscale intensity images without noise added.

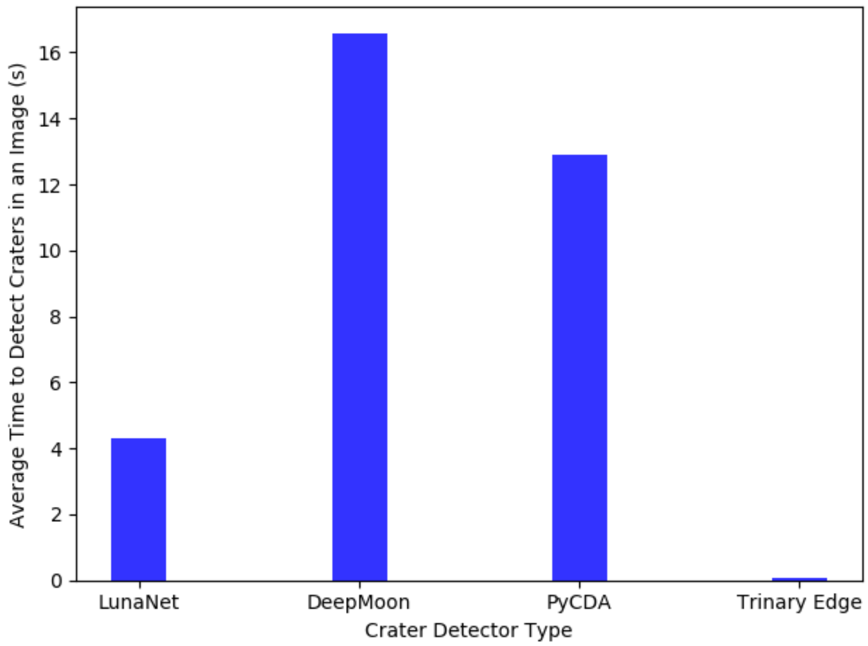


Fig. 10 Comparison of average length of time that it takes different crater detectors to process an input image and output a list of detected craters. LunaNet, DeepMoon and PyCDA were tested on the same hardware and framework. Trinary edge detector was tested on different hardware and framework. Detectors were tested on the same set of 100 LRO grayscale intensity images without noise added.

The performance of the crater detectors on the five sets of imagery is shown in Fig. 8. LunaNet produces the highest average number of good crater detections in all five image sets. DeepMoon's performance in comparison to LunaNet demonstrates that the additional CNN training and image processing of LunaNet effectively increased the crater detection activity and decreased the sensitivity to noise. PyCDA produces less than 50% of the average number of good detections that LunaNet does, with little sensitivity to noise effects. The trinary edge detector detects few known craters in each each image and demonstrates high levels of sensitivity to image exposure, which would make development and tuning a robust exposure adjustment algorithm critical for a lunar mission using the trinary edge detector. Some of the lower numbers of good crater matches produced by DeepMoon, PyCDA and LunaNet can be attributed to the strict requirement of detecting accurate crater rims. Examples of the four crater detectors' performances on images with the four different noise types are provided in the Appendix.

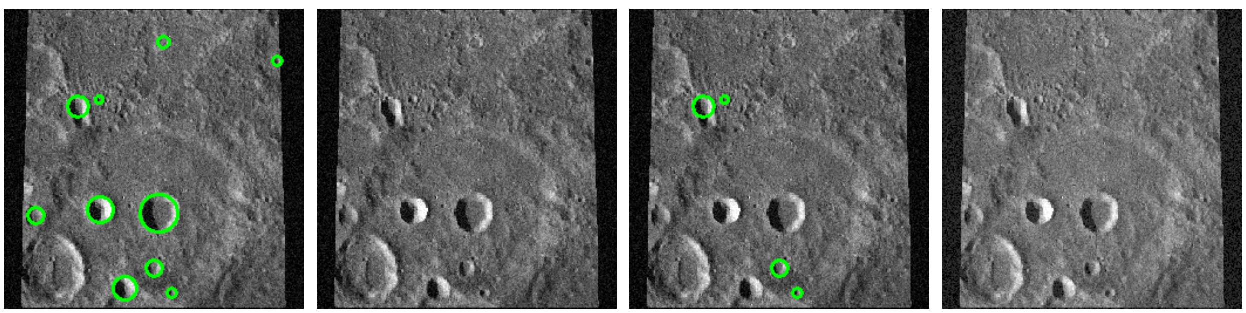
Fig. 9 shows information about the average accuracy error of crater detection from each of the four crater detectors. Location accuracy is the Euclidean distance error between the detected crater center and the known crater center that it is matched to, in pixels. Size accuracy is the error between the detected crater diameter and the known crater diameter that it is matched to, in pixels. LunaNet has the lowest error in both location and in size, indicating that it generates the highest accuracy crater detections of the four detectors. DeepMoon has more than four times the location and size error of LunaNet. PyCDA has close to double the location error and more than double the size error of LunaNet. The trinary edge detector performs similarly to PyCDA.

The time that it took each crater detector to process a lunar image and extract craters is also a relevant comparison since this system is targeted to be used for real-time lunar navigation. Fig. 10 shows a comparison of the average length of time that it takes each crater detector to process an input image and output a list of detected craters. LunaNet, DeepMoon and PyCDA were tested in Python Keras on an NVIDIA K80 GPU, while the trinary edge detector was tested in MATLAB R2016b on a 3.1 GHz Intel Core i5 MacBook Pro. The difference in framework and hardware makes it difficult to directly compare the computation time of the trinary edge detector with the others. These detector implementations were also in prototype forms and had not been optimized for real-time use. The trinary edge crater detector runs in a fraction of the amount of time as the other crater detectors. However, the trinary edge detector was tested in MATLAB, which is generally faster than Python. It is also relatively simplistic in its design, and as a consequence, it is not robust to changes in image qualities and detects a very low number of true craters even in ideal conditions. DeepMoon has the same neural network architecture as LunaNet, but it processes the neural network prediction with template matching. Template matching requires looping over all possible circle radii, contributing to its long run-time. PyCDA is made of two neural networks instead of just one, which adds to its run-time. LunaNet has shorter time requirements than other neural network-based crater detectors, but more than an image processing-based crater detector.

V. Conclusions

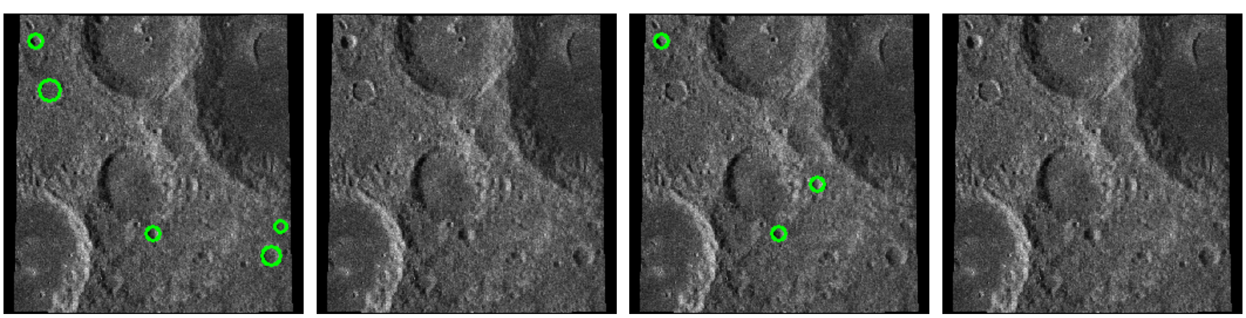
In conclusion, LunaNet produces at least twice as many good crater detections on average as DeepMoon, the neural network crater detector that it was based on, PyCDA, another neural network crater detector, and the trinary edge detector, a thresholding-based crater detector. Since a higher number of features corresponds to better navigation performance in terrain relative navigation, LunaNet appears to be a promising option for crater-based visual terrain relative navigation. In addition, the number of good crater detections that LunaNet produces is robust to changes in image quality due to Gaussian noise, shot noise, and brightness levels. This robustness to visual noise supports LunaNet's feasibility as a reliable navigation tool that does not require a high level of human input or tuning. Beyond crater matching, the ultimate goal of this system is to refine the position estimate of a spacecraft in-orbit and during landing. In order to test this performance, future work will include extensive simulations of a navigation filter that incorporates crater detections throughout a lunar orbit trajectory.

Appendix



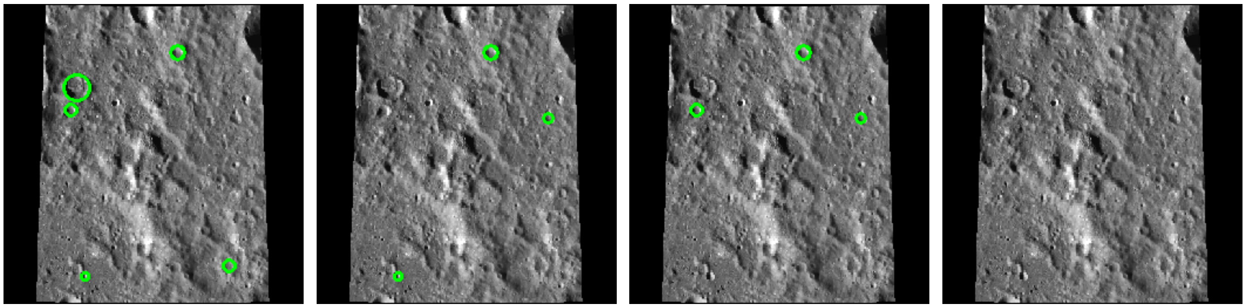
(a) LunaNet (b) DeepMoon (c) PyCDA (d) Trinary edge

Fig. 11 Craters that were detected by crater detectors and matched to known craters in a representative LRO image with Guassian noise added.



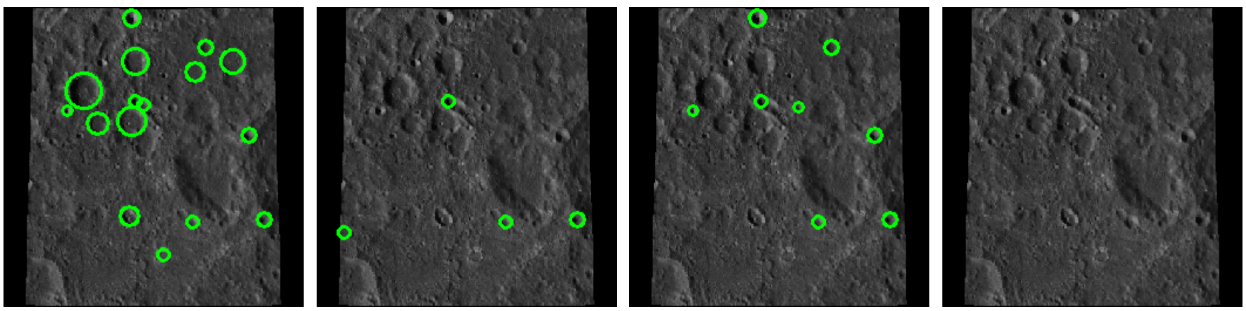
(a) LunaNet (b) DeepMoon (c) PyCDA (d) Trinary edge

Fig. 12 Craters that were detected by crater detectors and matched to known craters in a representative LRO image with shot noise added.



(a) LunaNet (b) DeepMoon (c) PyCDA (d) Trinary edge

Fig. 13 Craters that were detected by crater detectors and matched to known craters in a representative LRO image with 30% brightness increase.



(a) LunaNet (b) DeepMoon (c) PyCDA (d) Trinary edge

Fig. 14 Craters that were detected by crater detectors and matched to known craters in a representative LRO image with 30% brightness decrease.

References

[1] Li, S., Jiang, X., and Tao, T., "Guidance Summary and Assessment of the Chang'e-3 Powered Descent and Landing," *Journal of Spacecraft and Rockets*, Vol. 53, 2016, pp. 258–277. doi:10.2514/1.A33208.

[2] Barlow, N. G., Mest, S. C., Gibbs, V. B., and Kinser, R. M., "Compilation of a Global GIS Crater Database for the Moon," *AAS/Division for Planetary Sciences Meeting Abstracts #44*, 2012, p. 311.05.

[3] Singh, L., and Lim, S., "On Lunar On-Orbit Vision-Based Navigation: Terrain Mapping, Feature Tracking Driven EKF," *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Honolulu, Hawaii, 2008.

[4] Izzo, D., Märtens, M., and Pan, B., "A survey on artificial intelligence trends in spacecraft guidance dynamics and control," *Astrodynamics*, Vol. 3, No. 4, 2019, pp. 287–299. doi:10.1007/s42064-018-0053-6, URL `https://doi.org/10.1007/s42064-018-0053-6`.

[5] Mourikis, A. I., Trawny, N., Roumeliotis, S. I., Johnson, A. E., Ansar, A., and Matthies, L., "Vision-Aided Inertial Navigation for Spacecraft Entry, Descent, and Landing," *IEEE Transactions on Robotics*, Vol. 25, No. 2, 2009, pp. 264–80.

[6] Johnson, A. E., and Montgomery, J. F., "Overview of Terrain Relative Navigation approaches for precise lunar landing," *Proceedings of the IEEE Aerospace Conference*, Big Sky, Montana, 2008.

[7] Sibley, G., Matthies, L., and Sukhatme, G., "Sliding window filter with application to planetary landing," *Journal of Field Robotics*, Vol. 27, No. 5, 2010, pp. 587–608.

[8] Klear, M. R., "PyCDA: An Open-Source Library for Automated Crater Detection," *9th Planetary Crater Consortium*, Boulder, CO, 2018.

[9] Ronneberger, O., Fischer, P., and Brox, T., "U-Net: Convolutional Networks for Biomedical Image Segmentation," *CoRR*, Vol. abs/1505.04597, 2015. URL `http://arxiv.org/abs/1505.04597`.

[10] Silburt, A., Zhu, C., Ali-Dib, M., Menou, K., and Jackson, A., "DeepMoon: Convolutional neural network trainer to identify moon craters," Astrophysics Source Code Library, May 2018.

[11] Z. Povilaitis, R., S. Robinson, M., van der Bogert, C., Hiesinger, H., Meyer, H., and Ostrach, L., "Crater density differences: Exploring regional resurfacing, secondary crater populations, and crater saturation equilibrium on the moon," *Planetary and Space Science*, Vol. 162, 2018, pp. 41–51. doi:10.1016/j.pss.2017.05.006.

[12] Head, J. W., Fassett, C. I., Kadish, S. J., Smith, D. E., Zuber, M. T., Neumann, G. A., and Mazarico, E., "Global distribution of large lunar craters: implications for resurfacing and impactor populations," *Science*, Vol. 329, No. 5998, 2010, pp. 1504–1507.

[13] Sato, H., Robinson, M. S., Hapke, B., Denevi, B. W., and Boyd, A. K., "Resolved Hapke parameter maps of the Moon," *Journal of Geophysical Research: Planets*, Vol. 119, No. 8, 2014, pp. 1775–1805. doi:10.1002/2013JE004580, URL `https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2013JE004580`.

[14] Wagner, R. V. S., and E. J. Robinson, M. S., "New Mosaicked Data Products from the LROC Team," *46th Lunar and Planetary Science Conference*, 2015. URL `https://www.hou.usra.edu/meetings/lpsc2015/pdf/1473.pdf`.

[15] Peyerer, E., Robinson, M., Denevi, B., and the LROC Science Team, "Lunar Reconnaissance Orbiter Camera global morphological map of the Moon," *Lunar Planetary Science Conference*, 2011. URL `https://www.lpi.usra.edu/meetings/lpsc2011/pdf/2387.pdf`.

[16] Robinson, M. S., Brylow, S. M., Tschimmel, M., Humm, D., Lawrence, S. J., Thomas, P. C., Denevi, B. W., Bowman-Cisneros, E., Zerr, J., Ravine, M. A., Caplinger, M. A., Ghaemi, F. T., Schaffner, J. A., Malin, M. C., Mahanti, P., Bartels, A., Anderson, J., Tran, T. N., Eliason, E. M., McEwen, A. S., Turtle, E., Jolliff, B. L., and Hiesinger, H., "Lunar Reconnaissance Orbiter Camera (LROC) Instrument Overview," *Space Science Reviews*, Vol. 150, No. 1, 2010, pp. 81–124. doi:10.1007/s11214-010-9634-2, URL `https://doi.org/10.1007/s11214-010-9634-2`.

[17] Kingma, D. P., and Ba, J., "Adam: A Method for Stochastic Optimization," , 2014.

[18] Fischler, M. A., and Bolles, R. C., "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Commun. ACM*, Vol. 24, No. 6, 1981, pp. 381–395. doi:10.1145/358669.358692, URL `http://doi.acm.org/10.1145/358669.358692`.

[19] Mcgee, L. A., Schmidt, S. F., Mcgee, L. A., and Sc, S. F., "Discovery of the Kalman Filter as a Practical Tool for Aerospace and Industry," Tech. rep., National Aeronautics and Space Administration, Ames Research, 1985.

[20] Steiner, T. J., Brady, T. M., and Hoffman, J. A., "Graph-based terrain relative navigation with optimal landmark database selection," *2015 IEEE Aerospace Conference*, 2015, pp. 1–12. doi:10.1109/AERO.2015.7119053.

[21] Boyat, A. K., and Joshi, B. K., "A Review Paper: Noise Models in Digital Image Processing," *ArXiv*, Vol. abs/1505.03489, 2015.